தமிழ்நாடு அரசு

மேல்நிலை முதலாம் ஆண்டு

கணினி அறிவியல்

தொகுதி-II

தமிழ்நாடு அரசு விலையில்லாப் பாடநூல் வழங்கும் திட்டத்தின்கீழ் வெளியிடப்பட்டது

பள்ளிக் கல்வித்துறை

தீண்டாமை மனித நேயமற்ற செயலும் பெருங்குற்றமும் ஆகும்

XI Std Computer Science Vol-1 TM Introduction Pages.indd 1 03/09/18 5:28 PM

தமிழ்நாடு அரசு

முதல்பதிப்பு - *2018*

(புதிய பாடத்திட்டத்தின்கீழ்
வெளியிடப்பட்ட நூல்)

விற்பனைக்கு அன்று

பாடநூல் உருவாக்கமும்
தொகுப்பும்

மாநிலக் கல்வியியல் ஆராய்ச்சி மற்றும்
பயிற்சி நிறுவனம்
© SCERT 2018

நூல் அச்சாக்கம்

தமிழ்நாடு பாடநூல் மற்றும் கல்வியியல்
பணிகள் கழகம்

www.textbooksonline.tn.nic.in

முகப்புரை

மனித நாகரீக வளர்ச்சியின் மிக உயர்ந்த கண்டுபிடிப்பு "கணிப்பொறிகள்". கணிப்பொறிகள் நமது அன்றாட வாழ்வின் ஒவ்வொரு நிலையிலும், நீக்கமற நிறைந்து காணப்படுகிறது. இன்று நாம் வாழும் யுகம், "கணிப்பொறி யுகம்". இந்த யுகத்தில் கணிப்பொறி பற்றிய அறிவு இன்றியமையாத ஒன்றாகும். எவர் ஒருவர், கணிப்பொறியை இயக்கும் அடிப்படை அறிவை பெற்றிருக்கவில்லையோ அவர் எத்தனை பட்டங்கள் பெற்றிருப்பினும், கல்லாதவர் என்றே கருதப்படுவார். அந்த அளவுக்கு கணிப்பொறி கற்றல் அதி முக்கியத்துவம் வாய்ந்தது.

நாட்டின் வளர்ச்சி, இளைஞர்களின் கைகளில் தான் உள்ளது, ஒவ்வொரு இளைஞரும் கணிப்பொறி அறிவை பெறவேண்டியது அவசியமாகும். இதனை மனதில் கொண்டே, இந்த பாடப்புத்தகம் வடிவமைக்கப்பட்டுள்ளது.

இந்த பாடப்புத்தகத்தை எவ்வாறு பயன்படுத்துவது

- இந்த பாடப்புத்தகத்தை படிக்க கணிப்பொறி தொழில்நுட்பம் பற்றிய முன்னறிவு தேவையில்லை.
- ஒவ்வொரு அலகும், ஆசிரியர் மற்றும் மாணவர்கள் செய்து பழகக்கூடிய எளிய செயல்முறைகள் மற்றும் செயல்பாடுகளைக் கொண்டுள்ளது.
- தொழில்நுட்ப கலைச்சொற்களை எளிதில் புரிந்துகொள்வதற்காக, "கலைச்சொல் அகராதி" இணைக்கப்பட்டுள்ளது.
- "உங்களுக்குத் தெரியுமா?" பெட்டி செய்தி, கற்பவருக்கு பாடத் தொடர்பான கூடுதல் தகவல்களை தருகிறது.
- மென்பொருள் பயன்பாடுகளைப் பயன்படுத்தி, செய்முறை பயிற்சிகளைத் தீர்ப்பதற்கான "பயிற்சி பட்டறை" அறிமுகம் செய்யப்பட்டுள்ளது.
- QR குறியீடு, பாடத் தொடர்பான கூடுதல் தகவல்களை மின்கற்றல் மூலம் பெறும் வகையில் இணைக்கப்பட்டுள்ளது.

QR குறியீட்டை எவ்வாறு இணைப்பது?

o உங்கள் ஸ்மார்ட் கைபேசியில், QR ஸ்கேனர் பயன்பாட்டை கூகுள் பிளே ஸ்டோரிலிருந்து பதிவிறக்கம் செய்து, நிறுவிக்கொள்ளவும்.

o QR ஸ்கேனர் பயன்பாட்டை திறந்துகொள்ளவும்.

o ஸ்மார்ட் கைபேசியை பாடப்புத்தகத்திலுள்ள QR குறியீட்டுக்கு அருகில் கொண்டுசெல்லவும்.

o ஸ்மார்ட் கைபேசி, QR குறியீட்டை படித்தவுடன், குறியீட்டில் உள்ள இணைய முகவரி உங்கள் ஸ்மார்ட் கைபேசியில் இணைக்கப்பட்டு, பாடத் தொடர்பான கூடுதலான தகவல்களை அறிந்து கொள்ளலாம்.

QR GUIDE

XI Std Computer Science Vol-1 TM Introduction Pages.indd 3 03/09/18 5:28 PM

12ஆம் வகுப்பிற்கு பிறகு மேற்படிப்பு மற்றும் வேலை வாய்ப்பு ஆலோசனை

படிப்புகள்	கல்லூரிகள் / பல்கலைக் கழகங்கள்	வேலை வாய்ப்புகள்
	பொறியியல்	
இளங்கலை பொறியியல் (B.E./B.Tech)	இந்தியா மற்றும் வெளிநாடுகளில் உள்ள எல்லா பல்கலைக் கழகங்கள் மற்றும் அதன் உறுப்புக்கல்லூரிகள், சுயநிதிக்கல்லூரிகள்	மென்பொருள் பொறியாளர்/ வன்பொருள் பொறியாளர்/ மென்பொருள் உருவாக்குவர். கணிப்பொறி துறையில் மருத்துவ சேவைப் பிரிவு.
	அறிவியல் மற்றும் கலை	
இளங்கலை அறிவியல் (B.Sc.) – கணிப்பொறி அறிவியல் B.C.A. (இளங்கலை கணிப்பொறி பயன்பாட்டியல்) இளங்கலை அறிவியல் – கணிதம், இயற்பியல், வேதியியல், உயிரிவேதியியல், புவியியல், இதழியல்/ நூலக அறிவியல், அரசியல் அறிவியல், பயணம் மற்றும் சுற்றிலாவியல்.	இந்தியா மற்றும் வெளி நாடுகளில் உள்ள எல்லா பல்கலைக் கழகங்கள் மற்றும் அதன் உறுப்புக்கல்லூரிகள், சுயநிதிக்கல்லூரிகள்	அரசு மற்றும் தனியார் நிறுவனங்களில், நில அமைப்பியல் வல்லுனர், வணிக செயலாக்க புறத்திறனீட்டம் (B.P.O.)
	சட்டம்	
LLB, இளங்கலை– சட்டம், இளங்கலை மற்றும் சட்டம், இளங்கலை வணிகவியல், இளங்கலை வணிக மேலாண்மை, இளங்கலை வணிக நிர்வாகம்,	இந்தியா மற்றும் வெளி நாடுகளில் உள்ள எல்லா பல்கலைக் கழகங்கள் மற்றும் அதன் உறுப்புக்கல்லூரிகள், சுயநிதிக்கல்லூரிகள்	வழக்கறிஞர்கள், சட்ட அதிகாரிகள் மற்றும் அரசு வேலைகள்

படிப்புகள்	கல்லூரிகள் / பல்கலைக் கழகங்கள்	வேலை வாய்ப்புகள்
	வணிகவியல் படிப்புகள்	
இளங்கலை–வணிகவியல் (வரி மற்றும் வரி நடைமுறைகள்), இளங்கலை–வணிகவியல் (பயணம் மற்றும் சுற்றுலா), இளங்கலை–வணிகவியல் (வங்கிமேலாண்மை), இளங்கலை–வணிகவியல் (தொழில்முறை), இளங்கலை–மேலாண்மை நிர்வாகம். இளங்கலை–பங்குசந்தை, இளங்கலை–கணக்கியல் மற்றும் நிதி	இந்தியா மற்றும் வெளி நாடுகளில் உள்ள எல்லா பல்கலைக் கழகங்கள் மற்றும் அதன் உறுப்புக் கல்லூரிகள், சுயநிதிக்கல்லூரிகள்	தனியார் நிறுவனங்கள், வங்கிதுறைகள் மற்றும் சுயவேலை வாய்ப்புகள்.
	கணக்காயர் படிப்புகள்	
C.A. பட்டைய கணக்காயர், C.M.A. –செலவு மேலாண்மை கணக்காயர்	ICAI –இந்திய பட்டைய கணக்காயர் பயிற்சி நிறுவனம்,	கணக்காயர் தனியார் நிறுவனங்கள், வங்கி துறைகள் மற்றும் சுயவேலை வாய்ப்புகள்.
	அறிவியல் படிப்புகள்	
இளங்கலை–அறிவியல் தாவரவியல், விலங்கியல், மனையியல், உணவு மேலாண்மை, பால் தொழில் நுட்பம், உணவக மேலாண்மை, ஆடை அலங்கார மேலாண்மை, தகவல் தொடர்பு, முப்பரிமாண அசைவுபட தொழில்நுட்பம் (3D)	இந்தியா மற்றும் வெளி நாடுகளில் உள்ள எல்லா பல்கலைக் கழகங்கள் மற்றும் அதன் உறுப்புக் கல்லூரிகள், சுயநிதிக்கல்லூரிகள்	அரசு மற்றும் தனியார் நிறுவனங்களில், நில அமைப்பியல் வல்லுனர், வணிக செயலாக்க புறத்திறனீட்டம் (B.P.O.)

XI Std Computer Science Vol-1 TM Introduction Pages.indd 5 03/09/18 5:28 PM

பொருளடக்கம்

இயல் எண்	பாடத் தலைப்புகள்	பக்க எண்
	அலகு III – C++ ஓர் அறிமுகம்	
9	C++ ஓர் அறிமுகம்	1
10	பாய்வுக் கட்டுப்பாடு	54
11	C++ - ன் செயற்கூறுகள்	93
12	அணிகள் மற்றும் கட்டுருக்கள்	126
	அலகு IV – C++ பொருள் நோக்கு நிரலாக்க மொழி	
13	அறிமுகம் – பொருள்நோக்கு நிரலாக்க நுட்பங்கள்	178
14	இனக்குழுக்கள் மற்றும் பொருள்கள்	186
15	பல்லுருவாக்கம்	239
16	மரபுரிமம் (INHERITANCE)	261
	அலகு V – கணிப்பொறி நன்னெறி மற்றும் இணையப் பாதுகாப்பு	
17	கணிப்பொறி நன்னெறி மற்றும் இணையப் பாதுகாப்பு	305
18	கணிப்பொறியில் தமிழ்	319

HZFQ5V

மின்னூல்

HZHZHP

மதிப்பீடு

B137_11_CS_TM

இணைய வளங்கள்

QR GUIDE

பாடநூலில் உள்ள விரைவு குறியீட்டைப் (QR Code) பயன்படுத்துவோம்! எப்படி?

- உங்கள் திறன்பேசியில், கூகுள் playstore /ஆப்பிள் app store கொண்டு QR Code ஸ்கேனர் செயலியை இலவசமாகப் பதிவிறக்கம் செய்து நிறுவிக்கொள்க.
- செயலியைத் திறந்தவுடன், ஸ்கேன் செய்யும் பொத்தானை அழுத்தித் திரையில் தோன்றும் கேமராவை QR Code–இன் அருகில் கொண்டு செல்லவும்.
- ஸ்கேன் செய்வதன் மூலம் திரையில் தோன்றும் உரலியைச் (URL) சொடுக்க, அதன் விளக்கப் பக்கத்திற்குச் செல்லும்.

XI Std Computer Science Vol-1 TM Introduction Pages.indd 6 03/09/18 5:28 PM

அலகுIII C++ ஓர் அறிமுகம்

பாடம் 9

C++ ஓர் அறிமுகம்

MZPL7I

கற்றலின் நோக்கங்கள்:

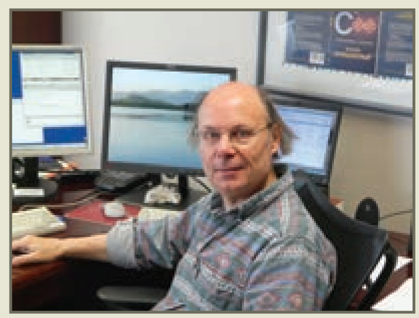
இந்தப் பாடப்பகுதியைக் கற்றபின் மாணவர் அறிந்து கொள்வது.

- C++ நிரலாக்க மொழியின் அடிப்படைக் கட்டுமான தொகுதியைப்பற்றி புரிந்து கொள்ளுதல்.
- எளிய C++ நிரல்களை உருவாக்க முடியும்.
- C++ நிரல்களை செயல்படுத்துதல் மற்றும் பிழைதிருத்துதல்.

9.1 முன்னுரை

C++ மொழி மிகவும் பிரபலமான நிரலாக்க மொழிகளில் ஒன்றாகும். இது 1979 ஆம் ஆண்டுகளின் முற்பகுதியில் ஏடி & டி (AT & T) பெல் ஆய்வுக் கூடத்தில் ஜேர்ன் ஸ்ட்ரௌஸ்ட்ரப் (Bjarne Stroustrup) அவர்களால் உருவாக்கப்பட்டது. C++ மொழி செயல்முறை மற்றும் பொருள் நோக்கு நிரலாக்க முறைகள் இரண்டையும் ஆதரிக்கிறது. எனவே, C++ ஒரு கலப்பின மொழி ஆகும். C++ அதன் முன்னோடி 'சி' மொழியின் நீட்டிப்பு ஆகும். ஜேர்ன் ஸ்ட்ரௌஸ்ட்ரப் தனது புதிய மொழியை "இனக்குழுக்களுடன் சி" என்று பெயரிட்டார். C++ என்னும் பெயரை (சி பிளஸ் பிளஸ் என உச்சரிக்க வேண்டும்) ரிக் மாஸ்சிட்டி (Rick Mascitti) என்பவர் சூட்டினார். இதில், C++ என்பது சி மொழியின் மிகுப்புச் செயற்குறி ஆகும்.

ஜேர்ன் ஸ்ட்ரெஸ்ரப் (Bjarne Stroustrup) C ++ நிரலாக்க மொழியை கண்டுபிடித்தவர்



ஜேர்ன் ஒரு டேனிஷ் கணிப்பொறி விஞ்ஞானி ஆவார். இவர் 1950 ஆம் ஆண்டு டிசம்பர் 30 ஆம் தேதி பிறந்தார். 1975 ஆம் ஆண்டில் டென்மார்க்கில் உள்ள ஆர்ஹஸ் பல்கலைக்கழகத்தில் (Aarhus University) கணிதம் மற்றும் கணிப்பொறி அறிவியலில் முதுகலை பட்டம் பெற்றார். 1979 ஆம் ஆண்டில் இங்கிலாந்தில் உள்ள கேம்பிரிட்ஜ் பல்கலைக்கழகத்தில் (University of Cambridge) கணிப்பொறி அறிவியலில் முனைவர் (Ph.D) பட்டம் பெற்றார்.

C ++ வரலாறு

C ++ ஆனது 1979 ஆம் ஆண்டில் AT &T பெல் ஆய்வகத்தில் ஜேர்ன் ஸ்ட்ரௌஸ்ட்ரப்(Bjarne Stroustrup) - ஆல் உருவாக்கப்பட்டது. C++ மொழி முதலில் C மொழியில் இருந்து தருவிக்கப்பட்டது மற்றும் சிமுலா(Simula), பி.சி.பி.எல் (BCPL), அடா(Ada), எம்.எல்.(ML), சி.எல்.

யூ(CLU) மற்றும் அல்கால் 68 (ALGOL 68) போன்ற பல மொழிகளால் தாக்கத்தை ஏற்படுத்தியது. இது "புதிய சி" மற்றும் "இனக்குழு உடன் சி" என்று 1983 வரை வழங்கப்பட்டது. 1983 ஆம் ஆண்டு, C ++ என ரிக் மாஸ்கிட்டியால்(Rick Mascitti.) பெயர் சூட்டப்பட்டது.

சர்வதேச தரநிர்ணய அமைப்பால் (ISO-International Organization for Standardization)-வால் சி ++ மொழி தரப்படுத்தப்பட்டது.

டிசம்பர் 2017 - ல் வெளியிடப்பட்ட சமீபத்திய நிலையான பதிப்பு ISO / IEC 14882: 2017 என அழைக்கப்படுகிறது, இது C++17 என அழைக்கப்படுகிறது. முதல் தரப்படுத்தப்பட்ட பதிப்பு 1998 இல் ISO / IEC 14882: 1998 ஆக வெளியிடப்பட்டது, பின்னர் C++ 03 (ISO / IEC 14882: 2003), C ++ 11 (ISO / IEC 14882: 2011) மற்றும் C ++ 14 (ISO / IEC 14882: 2014). அடுத்த நிலையான பதிப்பு 2020 இல் C++ 20 ஆக இருக்கும்.

C++ - ன் தாக்கம் C # (சி-ஷார்ப்) (C-Sharp)), டி(D), ஜாவா (Java) மற்றும் சி-இன் புதிய பதிப்புகளில் உள்ளது.

C++ -ன் நன்மைகள்

- C++ மிகவும் எளிமையான மொழியாகும், மேலும் பல்-சாதனம், பல்-பணித்தளம் பயன்பாட்டு வளர்ச்சிக்கு பெரும்பாலும் தேர்வு செய்யப்படும் மொழியாகும்.
- C++ ஒரு பொருள்நோக்கு நிரலாக்க மொழியாகும். இது இனக்குழுக்கள், மரபுரிமம், பல்லுருவாக்கம், தரவு அருவமாக்கம் மற்றும் உறைபொதியாக்கம் ஆகியவற்றை உள்ளடக்கியது.
- C++ களஞ்சியத்தில் நிறைய செயற்கூறுகள் உள்ளது.
- C++ விதிவிலக்கு கையாளுதலையும் **(exception handling)**, செயற்கூறு பணிமிகுப்பையும் அனுமதிக்கிறது. ஆனால் சி மொழியில் இந்த வசதி இல்லை.
- C+ + மொழி ஒரு சக்திவாய்ந்த, ஆற்றலுடைய மற்றும் விரைவான மொழியாகும். இது **GUI** பயன்பாடுகளிலிருந்து விளையாட்டுகளுக்கான **3D** வரைகளை நிகழ்நேர**(real-time)** கணித உருவகப்படுத்துதல்களுக்கு ஒரு பரவலான பயன்பாடுகளைக் கண்டறிகிறது.

9.2 குறியுருத் தொகுதி

- குறியுருத் தொகுதி என்பது ஒரு C++ நிரலை எழுதுவதற்கு அனுமதிக்கப்படும் எழுத்துகளின் தொகுப்பாகும். ஒரு எழுத்துரு என்பது பெரும்பாலும் எல்லா விசைப்பலகையிலும் உள்ள எழுத்து, எண் அல்லது குறியீடு (சிறப்பு குறியீடுகள்) ஆகும். பின்வரும் எழுத்துக்களை C++ ஏற்றுக்கொள்கிறது.

உங்களுக்குத் தெரியுமா?

சி மொழியை அடிப்படையாக சார்ந்த C++, ஜாவா, PHP போன்ற நிரலாக்க மொழிகளுக்கு குறியுரு தொகுதி, வில்லைகள் மற்றும் கோவைகள் ஆகியவை மிகவும் பொதுவானவை.

எழுத்துக்கள்	A …. Z, a ….. z
எண்	0 …. 9
சிறப்பு குறியீடு	+ - * / ~ ! @ # $ % ^& [ ] ( ) { } = >< _ \ | ? . , : ‘ “ ;
இடைவெளி	வெற்று இடைவெளி, கிடைமட்டத்தத்தல் (→), செலுத்தி திருப்பல்(↵), புதிய வரி, வரி செலுத்தி.
மற்ற குறியீடுகள்	C++ 256 ASCII உருக்களை தரவுகளாக செயல்படுத்தும்.

Chapter 9.indd 2 03/09/18 5:57 PM

9.3 மொழித் தொகுதி (வில்லைகள்)

C++ நிரல் கூற்றுகள், பல சிறிய கூறுகளால் நிர்வகிக்கப்படுகின்றன கட்டளைகள், மாறிகள், மாறிலிகள் மற்றும் செயற்குறிகள், நிறுத்தக் குறிகள் போன்ற குறியீடுகள் கொண்டு உருவாக்கப்படுகிறது. இந்த தனிப்பட்ட கூறுகள் கூட்டாக, மொழித்தொகுதிகள் (Lexical units) அல்லது மொழிக்கூறுகள் (Lexical elements) அல்லது வில்லைகள் (Tokens) என அழைக்கப்படுகின்றன. C++ பின்வரும் வில்லைகளை கொண்டுள்ளது:

சிறப்புச் சொற்கள் (Keywords)	குறிப்பெயர்கள் (Identifiers)
நிலையுருக்கள் (Literals)	செயற்குறிகள் (Operators)
நிறுத்தற்குறிகள்(Punctuators)	

குறிப்பு

வில்லைகள்

ஒரு நிரலில் உள்ள மீச்சிறு தனித்த அலகு, வில்லைகள் அல்லது மொழித் தொகுதி என்று அழைக்கப்படுகிறது.

9.3.1 சிறப்புச் சொற்கள் (Keywords)

C ++ -நிரல் பெயர்ப்பிக்கு மட்டுமே புரிகின்ற பொருள் கொண்ட காப்பு சொற்களுடன் (reserved words), அவை C++ நிரல்களைக் கட்டமைக்க, சிறப்புச் அவசியமான கூறுகளாகும். பெரும்பாலான சொற்கள் C, C ++ மற்றும் Java க்கு பொதுவானது.

குறிப்பு

C++ ஒரு எழுத்து வடிவ உணர்த்தி (case sensitive) மொழியாகும் எனவே, சிறப்புச் சொற்கள் சிறிய எழுத்துக்களில் இருக்க வேண்டும்.

அட்டவணை 9.1 C++ சிறப்புச் சொற்கள்

asm	auto	break	case	catch
char	class	const	continue	default
delete	do	double	else	enum
extern	float	for	friend	goto
if	inline	int	long	new
operator	private	protected	public	register
return	short	signed	sizeof	static
struct	switch	template	this	throw
try	typedef	union	un signed	virtual
void	volatile	while		

- திருத்தங்கள் மற்றும் சேர்த்தல்களுடன், பின் வரும் சிறப்புச் சொற்களும் அடங்கும்:

using, namespace, bal, static_cast, const_cast, dynamic_cast, true, false

- இரட்டை அடிக்கோடிடப்பட்ட கொண்ட குறிப்பெயர்கள், C++ செயலாக்கத்திற்கும் மற்றும் அடிப்படை களஞ்சியங்களைப் (standard libraries) பயன்படுத்துவதற்காக ஒதுக்கப்பட்டிருக்கின்றன. ஆகையால் இந்த குறிப்பெயர்களை பயனர்கள் தவிர்க்க வேண்டும்.

9.3.2 குறிப்பெயர்கள் (Identifiers)

குறிப்பெயர்கள் என்பது C++ நிரலில் வெவ்வேறு பகுதிகளுக்கு கொடுக்கப்படும் பெயர்களாகும். இவை பயனரால் வரையறுக்கப்பட்ட, மாறிகள், செயற்கூறுகள், அணிகள், இனக்குழுக்கள் போன்றவை ஆகும். இவை ஒரு நிரலின் அடிப்படை கட்டுமானத் தொகுதிகள் ஆகும். ஒவ்வொரு மொழியிலும் குறிப்பெயர்களுக்கு பெயரிடுவதற்கு என சில குறிப்பிட்ட விதிகள் உள்ளன.

குறிப்பெயர்களுக்கு பெயரிடுவதற்கான விதிகள்:

- ஒரு குறிப்பெயரின் முதல் எழுத்து கண்டிப்பாக எழுத்து அல்லது அடிக்கோடிட்டு (_) இருக்க வேண்டும்.
- எழுத்துக்கள், எண்கள் மற்றும் அடிக்கோடுகள்

Chapter 9.indd 3 03/09/18 5:57 PM

மட்டுமே அனுமதிக்கப்படுகின்றன. பிற சிறப்பு எழுத்துருக்கள் பெயரின் ஒரு பகுதியாக அனுமதிக்கப்படாது.

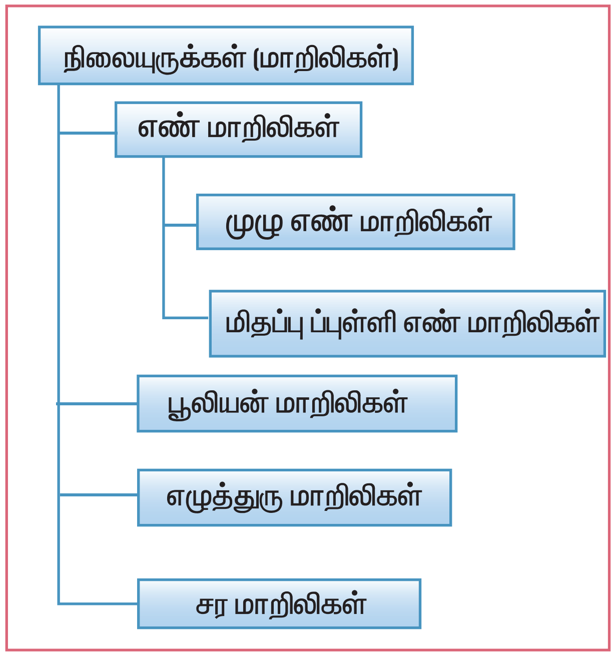
- C++ ஒரு எழுத்து வடிவ உணர்த்தியாகும் (case sensitive) பெரிய (Uppercase) மற்றும் சிறிய (lowercase) எழுத்துக்கள் வெவ்வேறாக கருதப்படுகின்றன.
- சிறப்புச் சொற்கள் (Keywords) அல்லது காப்புச் சொற்கள்(Reserve words) குறிப்பெயரின் பெயராக பயன்படுத்த முடியாது.
- ANSI தரநிலைகளின் படி, C++ -ல் குறிப்பெயர்களுக்கான எழுத்தின் நீளத்திற்கு எந்த எல்லையும் இல்லை. எனவே அனைத்து எழுத்துகளும் குறிப்பிடத்தக்கவை.

குறிப்பெயர்கள்	சரியா/ தவறா	தவறுக்கான காரணம்
Num	சரி	
NUM	சரி	
_add	சரி	
total_sales	சரி	
tamilMark	சரி	
num-add	தவறு	சிறப்பு எழுத்துரு (-) கொண்டுள்ளது
this	தவறு	இது ஒரு சிறப்புச் சொல்லை கொண்டுள்ளது. சிறப்புச் சொல்லை குறிப்பெயராக பயன்படுத்த முடியாது.
2myfile	தவறு	குறிப்பெயர் எப்பொழுதும் எழுத்து அல்லது அடிக்கோட்டுடன் தொடங்கவேண்டும்.

- மாறியின் பெயரை பல்வேறு பகுதிகளாக பிரிக்க, மாறியின் பெயர்களில் அடிக்கோடிட்டுக் கொள்ளலாம் (எ.கா: **total_sales** என்பது சரியான குறிப்பெயர்,**total sales** ஒரு தவறான குறிப்பெயர் ஆகும்.)
- **tamilMark** போன்ற பாணி கொண்ட மாறியை பயன்படுத்தலாம். இரண்டாவது வார்த்தையின் முதல் எழுத்து பெரிய எழுத்தில் இருப்பதைக் காணலாம்.

9.3.3 நிலையுருக்கள்(மாறிலிகள்) Literals (Constants)

- ஒரு நிரல் இயங்கும்போது மதிப்புகள் மாறாத தரவுகள் நிலையுருக்கள் ஆகும். எனவே, நிலையுருக்கள் மாறிலிகள் என அழைக்கப்படுகிறது. C++ -ல் பல வகையான நிலையுருக்கள் உள்ளன.



படம் 9.1 மாறிலிகளின் வகைகள்

எண் மாறிலிகள் (Numeric Constants)

மாறிகளாகப் பயன்படுத்தப்படும் எண்கள் எண் மாறிலிகள் ஆகும் . எண் மாறிலிகள் பின்வருமாறு வகைப்படுத்தப்பட்டுள்ளன:

1. முழு எண் மாறிலிகள் / நிலையான புள்ளி மாறிலிகள்(Integer Constants / Fixed point constants)

2. மெய் எண் மாறிலிகள் / மிதப்புப் புள்ளி மாறிலிகள் (Real constants / Floating point constants)

(1) முழு எண் மாறிலிகள் / நிலையான புள்ளி மாறிலிகள் (Integer Constants (or) Fixed point constants)

Chapter 9.indd 4 03/09/18 5:57 PM

பின்ன மதிப்புகள் இல்லாத எண்கள் முழு எண்களாகும். ஒரு முழு எண் மாறிலி எந்த ஒரு தசம புள்ளியும் இல்லாமல் குறைந்தபட்சம் ஒரு இலக்கத்தை கொண்டிருக்க வேண்டும். இது மறை குறியீடு அல்லது குறியீடு இல்லாமல் இருக்கும் மறை குறியீடு உள்ள முழு எண்கள் எதிர்மறை எண்ணாக கருதப்படுகின்றன. காற்புள்ளிகள் மற்றும் வெற்று இடைவெளிகள் ஆகியவை ஒரு பகுதியாக அனுமதிக்கப்படாது.

C++ இல், மூன்று வகையான முழுஎண் மாறிலிகள் உள்ளன: (i) பதின்மம் (ii) எண்ணிலை (iii) பதினாறு நிலை.

(i) பதின்மம் (Decimal)

0 முதல் 9 வரையான, ஒன்று அல்லது அதற்கு மேற்பட்ட முழு எண் இலக்கங்களின் வரிசை

சரி	தவறு
725	7,500 (காற்புள்ளி அனுமதியில்லை)
-27	66 5(வெற்று இடைவெளி அனுமதியில்லை)
4.56	9$ (சிறப்பி குறியீடு அனுமதியில்லை)

4.56 என்ற எண்ணை ஒரு முழுஎண் மாறியாக கொடுக்கும் போது, 4ஐ மட்டும் எடுத்துக்கொள்ளும், **.56** என்ற மதிப்பு நிராகரிக்கப்படும்.

குறிப்பு:

ஒரு பதின்ம மாறிலியை பின்ன மதிப்புகளுடன் கொடுத்தால், நிரல்பெயர்ப்பி முழுஎண் பகுதியை மட்டுமே எடுத்துக்கொள்ளும் பின்ன பகுதியை நிராகரித்துவிடும். இது உள்ளுறை இனமாற்றம் (Implicit Conversion) என அழைக்கப்படுகிறது. இதைப் பற்றி பின்னர் தெரிந்துகொள்ளலாம்.

(ii) எண்ம / எண்ணிலை (Octal)

ஒன்று அல்லது அதற்கு மேற்பட்ட எண்ம இலக்கங்களின் வரிசை (0 … .7). எண்ம மாறிலிகள் 0 என்ற முன்னொட்டுடன் தொடங்குகிறது.

சரி	தவறு
012	05,600(காற்புள்ளியை ஏற்காது)
-027	04.56 (தசம புள்ளியை ஏற்காது)**
+0231	0158 (8-எண்ம எண் இல்லை எனவே, இது ஏற்புடையதல்ல.)

குறிப்பு

**0ல் தொடங்கும் ஒரு பின்ன எண்ணை, எண்ணிலை எண்ணாக பயன்படுத்தும் போது, அது எண்ம எண்ணாக கருதாமல், பதின்ம எண்ணாக கருதப்படும்.

(iii) பதினாறுநிலை (Hexadecimal)

ஒன்று அல்லது அதற்கு மேற்பட்ட பதினாறுநிலை இலக்கங்களின் வரிசை (0 …. 9, A …. F). பதினாறுநிலை மாறிலிகள் 0x அல்லது 0X என்ற முன்னொட்டு உடன் தொடங்குகிறது.

சரி	தவறு
0x123	0x1,A5 (காற்புள்ளியை அனுமதிக்காது)
0X568	0x.14E (கொடுக்கப்பட்டுள்ளவாறு தசமபுள்ளியை அனுமதிக்காது)

பொதுவாக, ஒரு முழு எண்ணுடன், **L** அல்லது **l** மற்றும் **U** அல்லது **u** ஆகியவற்றுள் ஒன்றை பின்னொட்டாக கொடுக்கும் போது அது **long** அல்லது **unsigned** வகை மாறிலியாக கருதிக்கொள்ளும்.

(2) மெய் மாறிலிகள் / மிதப்புப்புள்ளி மாறிலிகள் (Real Constants (or) Floating point constants)

ஒரு மெய் அல்லது மிதப்புப்புள்ளி மாறிலி பின்னப்பகுதியை கொண்ட ஒரு எண் மாறிலி ஆகும். இந்த மாறிலிகள் பின்ன வடிவில் அல்லது அடுக்கு வடிவத்தில் எழுதப்படலாம்.

பின்ன வடிவில் உள்ள ஒரு மெய் மாறிலி குறியிடப்பட்ட (signed) அல்லது குறியிடப்படாத (unsigned), இலக்கங்களின் இடையில் ஒரு தசம புள்ளியைக் கொண்ட வரிசை ஆகும். ஒரு தசம

Chapter 9.indd 5 03/09/18 5:57 PM

புள்ளிக்கு முன்னும் பின்னும் ஒரு இலக்கத்தை அது கொண்டிருக்க வேண்டும். இதில் + அல்லது –குறியீடு முன்னொட்டாக இருக்கும். எந்த குறியீடும் இல்லாத மெய் மாறிலி நேர்மறை எண்ணாகக் கருதப்படுகிறது.

அடுக்கு (Exponent) வடிவில் உள்ள மெய் மாறிலிகள் இரண்டு பாகங்களைக் கொண்டுள்ளது: (1) அடிஎண் (Mantissa) மற்றும் (2) அடுக்குக்குறி (Exponent). அடிஎண் ஒரு முழுஎண் அல்லது மெய் மாறிலியாக இருக்கவேண்டும். ஒரு எண்ணை அடுக்குக்குறி வடிவில் கொடுப்பதற்கு அடிஎண்ணை தொடர்ந்து E அல்லது e என்ற ஒரு எழுத்து இருக்கவேண்டும். அடுக்குக்குறி எப்பொழுதும் முழுஎண்ணாக இருக்கவேண்டும்.

உதாரணமாக, 58000000.00 என்பதை 0.58×10^{8} அல்லது 0.58E8 என எழுதலாம்

அடிஎண்	அடுக்குகுறி
0.58	8

எடுத்துக்காட்டு

5.864 E-1 → 5.864×10^{1} → 58.64

5864 E-2 → 5864×10^{-2} → 58.64

0.5864 E-2 → 0.5864×10^{2} → 58.64

பூலியன் நிலையுருக்கள் (Boolean Literals)

பூலியன் நிலையுருக்கள் பூலியன் மதிப்புகளில் ஒன்றான மெய் அல்லது பொய் என்பதை குறிப்பிடப் பயன்படுத்தப்படுகின்றன. மெய் எனில் மதிப்பு 1-என்னும் மற்றும் பொய் எனில் மதிப்பு 0 –எனவும் எடுத்துக்கொள்ளும்.

குறியுரு மாறிலி (Character Constant)

குறியுரு மாறிலி என்பது ஒற்றை மேற்கோள் குறிகளுக்குள் தரப்படும் ஒற்றை குறியுருவைக் கொண்டிருக்கும். C++ -ல் ஒரு குறியுரு மாறிலி, ஒற்றை மேற்கோள் குறிகளுக்குள் ஒற்றை எழுத்தை கண்டிப்பாக கொண்டிருக்கவேண்டும்.

சரியான குறியுரு மாறிலிகள் : 'A', '2', '$'

தவறான குறியுரு மாறிலிகள் : "A"

ஒவ்வொரு ஒற்றை குறியுரு மாறிலி மதிப்புக்கும் நிகரான ASCII மதிப்பு உள்ளது. உதாரணமாக, 'A' -ன் மதிப்பு 65 ஆகும்.

விடுபடுவரிசை / வடிவற்ற-குறியுரு (Escape sequences / Non-graphic characters)

C++ சில அச்சிட முடியாத எழுத்துக்களை குறியுரு மாறிலிகளாக ஏற்கிறது. அச்சிட முடியாத எழுத்துக்கள், வடிவற்ற-குறியுருக்கள் என அழைக்கப்படுகின்றன. அச்சிட முடியாத எழுத்துக்கள் C++ இல் ஒரு நிரலை செயல்படுத்தும் போது, விசைப்பலகையிலிருந்து நேரடியாக தட்டச்சு செய்ய முடியாத எழுத்துக்கள் ஆகும்: உதாரணமாக back space, tabs போன்றவை. இந்த அச்சிட முடியாத எழுத்துக்கள் விடுபடுவரிசைகளை பயன்படுத்தி குறிப்பிடப்படுகின்றன. ஒரு விடுபடுவரிசை ஒரு பின்சாய்வுக் குறியீட்டால் குறிக்கப்படுகிறது, அதன் பின் ஒன்று அல்லது இரண்டு எழுத்துக்கள் வரலாம்.

அட்டவணை 9.2 விடுபடுவரிசை

விடுபடுவரிசை	வடிவற்ற குறியுரு
\a	மணி ஒலிப்பு
\b	பின்னிட வெளி
\f	பக்கம் செலுத்தி
\n	புதிய வரி / வரிச்செலுத்தி
\r	செலுத்தித் திருப்பல்
\t	கிடைமட்டத் தத்தல்
\v	செங்குத்துத் தத்தல்
****	பின்சாய்வுக் கோடு
\'	ஒற்றை மேற்கோள்
\"	இரட்டை மேற்கோள்
\?	கேள்விக்குறி
\On	எண்ம / எண்ணிலை எண்
\xHn	பதினறும / பதினாறுநிலை எண்
\0	மதிப்பிலி

ஒரு விடுபடுவரிசை இரண்டு எழுத்துகளைக் கொண்டிருந்தாலும், அவை ஒற்றை மேற்கோள் குறிக்குள் குறிப்பிடப்பட வேண்டும். ஏனெனில், C++ விடுபடுவரிசையை ஒரு குறியுரு மாறிலியாக கருதுகிறது மற்றும் ASCII குறியீட்டு முறையில், நினைவகத்தில் ஒரு பைட் மட்டுமே ஒதுக்கீடு செய்கிறது.

உங்களுக்குத் தெரியுமா?

ASCII (American Standard Code for Information Interchange) முதன் முதலில் 1963 ஆம் ஆண்டில் X3 கமிட்டி மூலமாக உருவாக்கப்பட்டது, இது American Standards Association (ASA) இன் ஒரு பிரிவாகும்.

சரநிலையுருக்கள் (String literals)

சரநிலையுருக்கள் என்பது இரட்டை மேற்கோள் குறிகளுக்குள் தரப்படும் குறியுருக்களின் வரிசையைக் கொண்டிருக்கும். சரநிலையுருக்கள்தானமைவாக '\0' என்னும் சிறப்புக் குறியுருவை ஈற்றில் இணைத்துக்கொள்ளும். எனவே "welcome" என்ற சரம் "welcome\0" என்று நினைவகத்தில் குறிக்கப்படுகிறது மற்றும் இந்த சரத்தின் அளவு 7 என்று குறிக்கப்படாமல் 8 குறியுருக்களாக குறிக்கப்படும். அதாவது, ஈற்றில் இணைக்கப்படும் '\0' வையும் சேர்த்துக் கொள்ளும்.

சரியான சரநிலையுருக்கள்: "A", "Welcome" "1234"

தவறான சரநிலையுருக்கள்: 'Welcome', '1234'

சுய மதிப்பீடு

1. நிலையுரு என்றால் என்ன? C++ ல் முழு நிலையுருக்களின் வகைகள் யாவை?
2. பின்வருவன எவ்வகையான மாறிலிகள்?

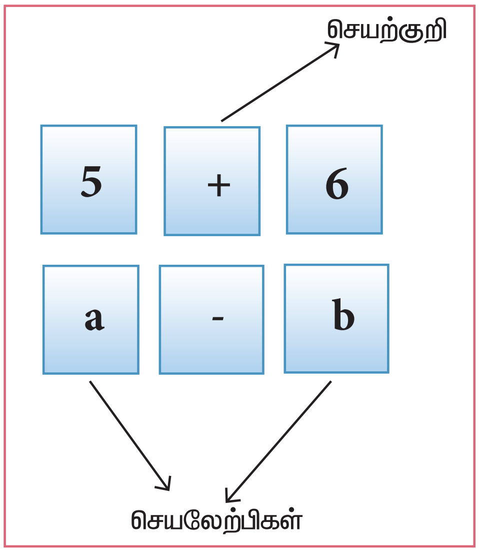
   (அ) 26 ஆ) 015 இ) 0xF ஈ) 0149
3. C++ல் குறியுரு மாறிலி என்றால் என்ன?
4. C++ல் விடுபடுவரிசை எவ்வாறு குறிக்கப்படுகிறது?
5. பின்வரும் மெய் மாறிலிகளை படி அடுக்கு முறையில் எழுதுக.

   (அ) 32.179 ஆ) 8.124 இ) 0.00007
6. பின்வரும் மெய் மாறிலிகளை மிதப்புப் புள்ளி எண் வடிவம் எழுதுக.

   (அ) 0.23E4 ஆ)0.517E-3 இ) 0.5E-5
7. சரத்தில் மதிப்பிலி (\0) குறியுருவின் முக்கியத்துவம் என்ன?

9.3.4 செயற்குறிகள் (Operators)

செயற்குறிகள் என்பது சில கணித மற்றும் ஏரண செயல்பாடுகளை செய்ய பயன்படும் குறியீடுகளாகும். செயலேற்பிகள் **(Operands)** என்பது செயற்குறிகளால் செயல்படுத்தப்படும் தரவு கூறுகள் அல்லது மதிப்புகளை குறிக்கிறது.



C++ -ல், செயலேற்பிகளின் எண்ணிக்கையின் அடிப்படையில் செயற்குறிகள் பின்வருமாறு வகைப் படுத்தப்படுகின்றன.

(i) ஒரும செயற்குறிகள் (Unary Operators) - ஒரே ஒரு செயலேற்பியை மட்டும் ஏற்கும்

(ii) இரும செயற்குறிகள் (Binary Operators) - இரண்டு செயலேற்பியை மட்டும் ஏற்கும்

(iii) மும்ம செயற்குறிகள் (Ternary Operators) - மூன்று செயலேற்பியை மட்டும் ஏற்கும்

பொதுவாக, C++ செயற்குறிகள் பின்வருமாறு வகைப்படுத்தப்படுகின்றன.

1. கணக்கீட்டுச்செயற்குறிகள் (Arithmetic Operators)
2. ஒப்பீட்டுச்செயற்குறிகள் (Relational Operators)
3. தருக்கச்செயற்குறிகள் (Logical Operators)
4. பிட் நிலைசெயற்குறிகள் (Bitwise Operators)
5. மதிப்பிருத்து செயற்குறிகள் (Assignment Operators)
6. நிபந்தனைச்செயற்குறிகள் (Conditional Operator)
7. பிற செயற்குறிகள் (Other Operators)

(1) கணக்கீட்டுச் செயற்குறிகள் (Arithmetic Operators)

கணக்கீட்டுச் செயற்குறிகள் எளிய கணிதச் செயல்பாடுகளாகிய கூட்டல், கழித்தல், பெருக்கல், மற்றும் வகுத்தல் போன்ற கணக்கீடுகளை செயல்படுத்துகிறது.

செயற்குறிகள்	செயல்பாடு	எடுத்துக்காட்டு
+	கூட்டல்	10 + 5 = 15
-	கழித்தல்	10 – 5 = 5
*	பெருக்கல்	10 * 5 = 50
/	வகுத்தல்	10 / 5 = 2 (வகுத்தலின் ஈவு)
%	முழு எண் வகுமீதி (வகுத்தலின் மீதியை கண்டுபிடிக்கும்)	10 % 3 = 1(வகுத்தலின் மீதி)

- மேலே குறிப்பிடப்பட்டுள்ள கணக்கீட்டுச் செயற்குறிகள் இரும செயற்குறிகளாகும். இருமச் செயற்குறிகளுக்கு குறைந்தபட்சம் இரண்டு செயலேற்பிகள் தேவை.

மிகுப்பு மற்றும் குறைப்புச்செயற்குறிகள் Increment and Decrement Operators

+ + (Plus, Plus) மிகுப்பு செயற்குறி

-- (Minus, Minus) குறைப்புச் செயற்குறி

மிகுப்பு அல்லது குறைப்பு செயற்குறி ஒரு செயலேற்பியின் மீது மட்டுமே செயற்பட்டு ஒரு புதிய மதிப்பை வழங்குகிறது. எனவே, இந்தச் செயற்குறிகள் ஒருமச் செயற்குறிகள் எனப்படும். மிகுப்பு செயற்குறி செயலேற்பியுடன் 1 என்ற மதிப்பை கூட்டுகிறது. குறைப்புச் செயற்குறி செயலேற்பியிலிருந்து 1 என்ற மதிப்பை குறைக்கிறது.

எடுத்துக்காட்டாக

- **x++** என்பது **x=x +1** என்பதற்குச் சமம். →**x** -ன் தற்போதைய மதிப்புடன் **1**ஐ கூட்டுகிறது.
- **x--** என்பது **x= x-1** என்பதற்குச் சமம். → **x** -ன் தற்போதைய மதிப்புடன் 1ஐ குறைக்கிறது.

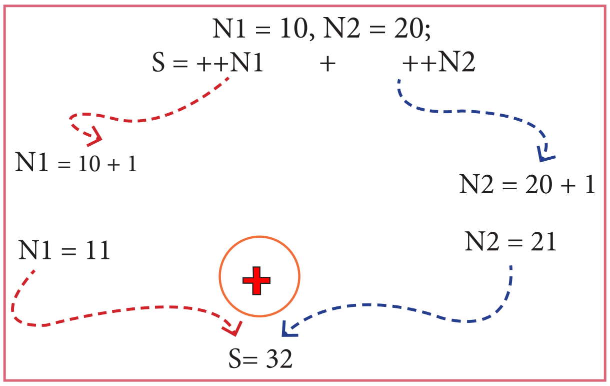
இந்த மிகுப்பு (++) அல்லது குறைப்பு (--) செயற்குறிகள் மாறிகளுக்கு முன்னொட்டாகவோ அல்லது பின்னொட்டாகவோ இருத்தப்படுகிறது. முன்னொட்டு முறையில் செயலேற்பியை பயன்படுத்துவதற்கு முன்னதாகவே மிகுப்பு அல்லது குறைப்பு செயல்படுத்தப்படும்.

எடுத்துக்காட்டாக: **N1=10, N2=20;**

S = ++N1 + ++N2;

Chapter 9.indd 8 03/09/18 5:57 PM

பின்வரும் வரைபடம் மேலே கொடுக்கப்பட்டுள்ள கூற்றின், செயல்படும் முறையை விளக்குகிறது.



படம் 9.2 முன்னொட்டு மிகுப்பு செயல்படும் முறை

மேலே உள்ள எடுத்துக்காட்டில் N-களின் மதிப்புகள் முதலில் 1 மிகுக்கப்பட்டு அதற்குரிய செயலேற்பியில் இருத்தப்படுகிறது.

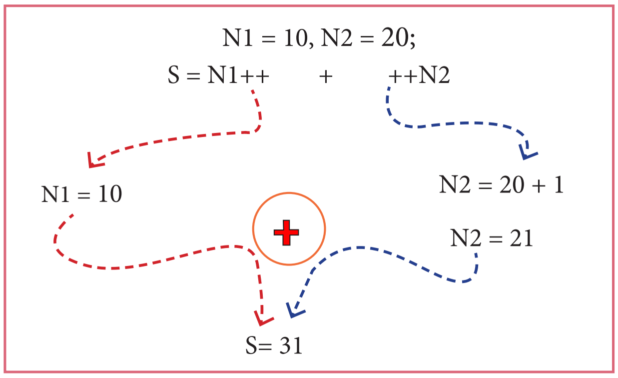
பின்னொட்டு முறையில், மிகுப்பு / குறைப்புக்கு முன்னதாகவே செயலேற்பியில் உள்ள மதிப்பை, கணக்கிடுதலுக்காக C++ பயன்படுத்துகிறது.

எடுத்துக்காட்டாக

N1=10, N2=20;

S = N1++ + ++N2;

பின்வரும் வரைபடம் மேலே கொடுக்கப்பட்டுள்ள கூற்றின், செயல்படும் முறையை விளக்குகிறது.



படம் 9.3 பின்னொட்டு மிகுப்பு செயல்படும் முறை

மேலே கொடுக்கப்பட்டுள்ள எடுத்துக்காட்டில் N1 செயலேற்பியில் இருத்தப்பட்ட மதிப்பு எடுத்துக்கொள்ளப்பட்டு, பிறகு அந்த மதிப்பானது மிகுக்கப்படுகிறது. (மதிப்பு 1கூட்டப்படுகிறது.)

செயற்குறி	செயல்பாடு	எடுத்துக்காட்டு (n=2 எனில், a ன் மதிப்பு என்ன?)	
		முன்னொட்டு	பின்னொட்டு
++	மிகுப்பு	a=++n;	a=n++;
		a-ன் மதிப்பு 3 (a = 3)	a-ன் மதிப்பு 2 (a = 2)
--	குறைப்பு	a=--n;	a=n--;
		a-ன் மதிப்பு 1 (a = 1)	a-ன் மதிப்பு 2 (a = 2)

(2) ஒப்பீட்டுச் செயற்குறிகள் (Relational Operators)

ஒப்பீட்டுச் செயற்குறிகள் செயலேற்பிகளுக்கு இடையேயான உறவு முறையை கண்டுபிடிக்க பயன்படுகிறது. ஒப்பீட்டுச் செயற்குறிகள் இரண்டு செயலேற்பிகள் மீது செயல்படுத்தப்படும்போது, விடையானது பூலியன் மதிப்பாக இருக்கும். 1 அல்லது 0 என்பது முறையே சரி அல்லது தவறு என்பதைக் குறிக்கிறது. C++, ஆறு ஒப்பீட்டுச் செயற்குறிகளை வழங்குகிறது. அவைகள்,

செயற்குறி	செயல்பாடு	எடுத்துக்காட்டு
>	விடப்பெரிது	a > b
<	விடச் சிறிது	a < b
>=	விடப்பெரிது அல்லது நிகர்	a >= b
<=	விடச் சிறிது அல்லது நிகர்	a <= b
==	நிகரானது	a == b
!=	நிகரில்லை	a != b

மேலே கொடுக்கப்பட்டுள்ள எடுத்துக்காட்டில் செயலேற்பி a-செயலேற்பி b-யுடன் ஒப்பிடப்பட்டு ஒப்பீட்டின் அடிப்படையில் விடை 1 அல்லது 0 என வரும். அதாவது, 1 என்பது சரி, 0 என்பது தவறு.

Chapter 9.indd 9 03/09/18 5:57 PM

- அனைத்து ஆறு ஒப்பீட்டு செயற்குறிகளும் இருமச் செயற்குறிகள் ஆகும்.

(3)தருக்கச் செயற்குறிகள் (Logical Operators)

தருக்கச் செயற்குறிகள், தருக்க மற்றும் ஒப்பீட்டு கோவைகளை மதிப்பிட பயன்படுகிறது. தருக்க செயற்குறிகள் செயலேற்பிகளாகிய தருக்க கோவைகளின் மீது செயல்படுகிறது. C++ மூன்று தருக்கச் செயற்குறிகளை வழங்குகிறது.

அட்டவணை 9.3 தருக்கசெயற்குறிகள்

செயற்குறி	செயல்பாடு	விளக்கம்
&&	**AND**	தருக்க AND இரண்டு வேறுபட்ட ஒப்பீட்டு கோவைகளை ஒன்றாக இணைக்கிறது. இரண்டு கோவைகளும் சரியெனில் 1 (True) என்ற மதிப்பை தரும். இல்லை எனில் 0 (False) என்ற மதிப்பை தரும்..
||	**OR**	தருக்க OR இரண்டு வேறுபட்ட ஒப்பீட்டு கோவைகளை ஒன்றாக இணைக்கிறது. ஏதேனும் ஒரு கோவை சரியெனில் 1 (True) என்ற மதிப்பை தரும். இரண்டு கோவைகளும் தவறு எனில் 0 (False) என்ற மதிப்பை தரும்.
!	**NOT**	NOT ஒரு கோவை அல்லது ஒரு செயலேற்பியின் மீது செயல்படுகிறது. இது மெய் மதிப்பை மாற்றி அல்லது தலைகீழாக கொடுக்கும். அதாவது, ஒரு செயலேற்பி அல்லது கோவை 1 (True) எனில் இந்த செயற்குறியானது 0 (False) என்னும் மதிப்பை தரும். 0 (False) எனில் 1 (True) மதிப்பை தரும்.

- AND, OR இரண்டும் இரும செயற்குறிகள் NOT என்பது ஒருமசெயற்குறி ஆகும்.

எடுத்துக்காட்டு: a = 5, b = 6, c = 7;

கோவை	விடை
(a<b) && (b<c)	1 (சரி - True)
(a>b) && (b<c)	0 (தவறு - False)
(a<b) || (b>c)	1 (சரி - True)
!(a>b)	1 (சரி - True)

(4)பிட்நிலை செயற்குறிகள் (Bitwise Operators)

பிட்நிலை செயற்குறிகள், தரவின் ஒவ்வொரு பிட்டின் (bit-by-bit) அடிப்படையிலும் செயல்படுகிறது. C++ இல், மூன்று வகையான பிட்நிலை செயற்குறிகள் உள்ளன. அவை:

(i) தருக்க பிட்நிலை செயற்குறிகள் (Logical bitwise operators)

(ii) பிட்நிலை நகர்வு செயற்குறிகள் (Bitwise shift operators)

(iii) பிட்நிலை ஒன்றின் நிரப்பு செயற்குறிகள் (One's compliment operators)

(i) தருக்க பிட்நிலை செயற்குறிகள் (Logical bitwise operators):

& பிட்நிலை **AND (இரும AND)**

| பிட்நிலை **OR (இரும OR)**

^ பிட்நிலை தனித்த **OR (இரும XOR)**

- பிட்நிலை AND (&) எனும் செயற்குறி இரண்டு செயலேற்பிகளின் மதிப்புகளும் 1 (True) என இருக்கும்போது 1 (True) எனவும் இல்லையேல் 0 (False) எனவும் விடை தரும்.

- பிட்நிலை OR (|) எனும் செயற்குறி, ஏதேனும் ஒரு செயலேற்பின் மதிப்பு 1 (True) என இருக்கும்போது 1 (True) என்ற விடையையும், இரண்டு செயலேற்பிகளின் மதிப்புகளும் 0 (False) என இருக்கும்போது 0 (False) என்ற விடையையும் தரும்.
- பிட்நிலை XOR (^) எனும் செயற்குறி ஒரே ஒரு செயலேற்பின் மதிப்பு 1 (True) என இருக்கும்பொழுது 1 (True) என்ற விடையையும், இரண்டு செயலேற்பிகளும் (1 (True) அல்லது 0 (False)) என சமமாக இருக்கும்பொழுது 0 (False) என்ற விடையையும் தரும்.

AND, OR, XOR –க் கான பிட்நிலை செயற்குறிகளின் மெய்ப்பட்டியல்(Truth table)

<table>
<tr><th>A</th><th>B</th><th>A & B</th><th>A | B</th><th>A ^ B</th></tr>
<tr><td>1</td><td>1</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
</table>

எடுத்துக்காட்டு

a = 65, b = 15 எனில் அதன் இருநிலை மதிப்புகள்

65 = 0100 0001

15 = 0000 1111

<table>
<tr><th>செயற்குறி</th><th>செயல்பாடு</th><th colspan="9">விடை</th></tr>
<tr><td rowspan="4">&</td><td rowspan="4">a & b</td><td>a</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>b</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>a & b</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td colspan="9">(a&b) = 0000 0001$_2$= 1$_{10}$</td></tr>
<tr><td rowspan="4">|</td><td rowspan="4">a | b</td><td>a</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>b</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>a | b</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td colspan="9">(a|b) = 01001111$_2$= 79$_{10}$</td></tr>
<tr><td rowspan="4">^</td><td rowspan="4">a ^ b</td><td>a</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>b</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>a ^ b</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td colspan="9">(a^b) = 0100 1110$_2$= 78$_{10}$</td></tr>
</table>

(ii) பிட்நிலை நகர்வு செயற்குறிகள் (The Bitwise shift operators):

C++ இல் இடது நகர்வு (<<) மற்றும் வலது நகர்வு (>>) எனும் இரண்டு பிட்நிலை நகர்வு செயற்குறிகள் உள்ளன.

இடது நகர்வு (shift left)(<<) – வலது செயலேற்பியில் கொடுக்கப்படும் முழு எண் மதிப்பின் எண்ணிக்கையின் அளவுக்கு, இடது செயலேற்பியின் பிட்டுகளை இடது பக்கமாக நகர்த்தும். வலது செயலேற்பி unsigned முழு எண்ணாக இருத்தல் வேண்டும்.

Chapter 9.indd 11 03/09/18 5:57 PM

வலது நகர்வு (shift right) (>>) – வலது செயலேற்பியில் கொடுக்கப்படும் முழு எண் மதிப்பின் எண்ணிக்கையின் அளவுக்கு, இடது செயலேற்பியின் பிட்டுகளை, வலது பக்கமாக நகர்த்தும். வலது செயலேற்பி unsigned முழு எண்ணாக இருத்தல் வேண்டும்.

எடுத்துக்காட்டு

a = 15 எனில் அதன் இருநிலை மதிப்பு 0000 1111

<table>
<tr><th>செயற்குறி</th><th>செயல்பாடு</th><th colspan="9">விடை</th></tr>
<tr><td rowspan="3"><<</td><td rowspan="3">a << 3</td><td>a</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>a << 3</td><td>0</td><td>1</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td colspan="9">$(a<< 3) = 01111000_2 = 120_{10}$</td></tr>
<tr><td rowspan="3">>></td><td rowspan="3">a >> 2</td><td>a</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>a>> 2</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td colspan="9">$(a>> 2) = 0000\ 0011_2 = 3_{10}$</td></tr>
</table>

(iii) பிட்நிலை ஒன்றின் நிரப்பு செயற்குறி (Bitwise one's compliment operator):

பிட்நிலை ஒன்றின் நிரப்பு செயற்குறி ~ (Tilde) இருநிலை எண்ணில் உள்ள பிட்டு 1 எனில் 0 எனவும், 0 எனில் 1 எனவும் மாற்றும். இது ஒரு ஒரும செயற்குறி) ஆகும்.

எடுத்துக்காட்டு

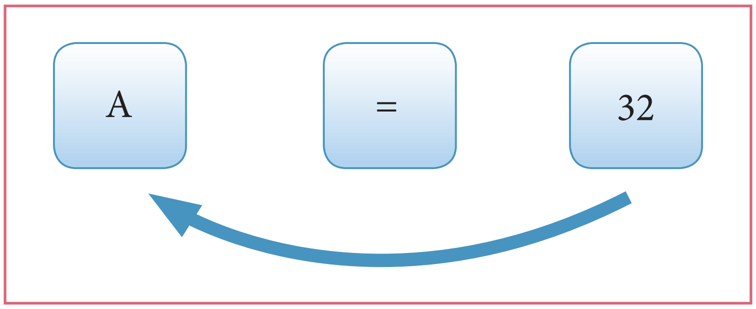
a = 15 எனில் அதன் இருநிலை மதிப்பு **0000 1111**

<table>
<tr><th>செயற்குறி</th><th>செயல்பாடு</th><th colspan="9">விடை</th></tr>
<tr><td rowspan="3">~</td><td rowspan="3">(~a)</td><td>a</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>(~a)</td><td>1</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td colspan="9">$(\sim a) = 1111\ 0000_2 = -16_{10}$</td></tr>
</table>

5. மதிப்பிருத்து செயற்குறி (Assignment Operator):

செயற்குறி = (சமம்) என்பது சாதாரண மதிப்பிருத்து செயற்குறி ஆகும். ஒரு மதிப்பிருந்து கூற்றின் வலப்பக்கம் இருக்கும் மதிப்பை இடப்பக்கம் உள்ள மாறியில் இருத்தும். இது பொதுவாக எல்லா கணிப்பொறி மொழிகளிலும் பயன்பாட்டில் உள்ளது. இது ஒரு இரும செயற்குறி ஆகும்.

Chapter 9.indd 12 03/09/18 5:57 PM



C++ பல விதமான மதிப்பிருந்து செயற்குறிகளைப் பயன்படுத்துகிறது. அவைகள் குறுக்கு வழி மதிப்பிருத்து செயற்குறிகள் எனப்படும்.

செயற்குறி	செயற்குறியின் பெயர்	எடுத்துக்காட்டு
+=	கூட்டல் மதிப்பிருத்து (Addition Assignment)	a = 10; c = a += 5; (ie, a = a + 5) c = 15
-=	கழித்தல் மதிப்பிருத்து (Subtraction Assignment)	a = 10; c = a -= 5;(ie. a = a – 5)c = 5
*=	பெருக்கல் மதிப்பிருத்து (Multiplication Assignment)	a = 10; c = a *= 5; (ie. a = a * 5) c = 50
/=	வகுத்தல் மதிப்பிருத்து (Division Assignment)	a = 10; c = a /= 5; (ie. a = a / 5) c = 2
%=	வகுமீதி மதிப்பிருத்து (Modulus Assignment)	a = 10; c = a %= 5; (ie. a = a % 5) c = 0

= மற்றும் == செயற்குறிகளுக்கு இடையே உள்ள வேறுபாட்டை விவரி.

(6) நிபந்தனைச் செயற்குறி (Conditional Operator):

C++இல் உள்ள ஒரே ஒரு நிபந்தனைச் செயற்குறி (?:) உள்ளது. இது ஒரு மும்ம செயற்குறி ஆகும். இந்த செயற்குறி if … else கட்டுப்பாட்டு கூற்றுக்கு மாற்றாகப் பயன்படுகிறது. இதைப் பற்றி பின்வரும் if … else கூற்றுயுடன் விரிவாகக் காண்போம்.

(7) பிற செயற்குறிகள் (Other Operators):

காற்புள்ளி (comma)	(,) என்னும் செயற்குறி கோவைகளின் தொகுப்பை இடது புறத்தில் இருந்து வலதுபுறமாக பிரித்து மதிப்பிடுகிறது.
Sizeof	இது ஒரு தொகுப்பு நேர செயற்குறி (compile time operator). இது மாறியின் அளவை பைட்டுகளில் கொடுக்கிறது.

Chapter 9.indd 13 03/09/18 5:57 PM

சுட்டுமாறி (Pointer)	* மதிப்பு சுட்டல் (Pointer to a variable) & முகவரி சுட்டல்(Address of)
பொருட்கூறு தெரிவு (Component selection)	. நேரடிப் பொருட்கூறு தெரிவு **(Direct component selector)** -> மறைமுகப் பொருட்கூறு தெரிவு **(Indirect component selector)**
இனக்குழு உறுப்பினர் செயற்குறி (Class member operators)	:: வரையெல்லை அணுகல் / கூறுபாட்டுச் செயற்குறி **(Scope access / resolution)** .* குறிப்பு விலக்குச் செயற்குறி **(Dereference)** ->* இனக்குழு உறுப்புக்கான குறிப்பு விலக்கம் **(Dereference pointer to class member)**

செயற்குறிகளின் முன்னுரிமை **(Precedence of Operators):**

செயற்குறிகள் முன்னுரிமையின் அடிப்படையில் செயல்படுத்தப்படுகின்றன. செயலேற்பிகளும், செயற்குறிகளும் குறிப்பிட்ட தருக்க முறைப்படி குழுவாக்கப்பட்டு மதிப்பிடப்படுகின்றன. இத்தகைய குழு வாக்கம் தொடர்புறுத்தம் **(Association)** எனப்படுகிறது.

முன்னுரிமை வரிசை (The order of precedence):

( ) [ ]	அடைப்புக்குறிக்குள் உள்ள செயற்குறிகள் முதலில் நிறைவேற்றப்படுகிறது.	மேல்
++, --	பின்னொட்டு மிகுப்பு / குறைப்பு	↓
++, --	முன்னொட்டு மிகுப்பு / குறைப்பு	
*, /, %	பெருக்கல், வகுத்தல், முழுஎண் வகுமீதி	
+, -	கூட்டல், கழித்தல்	
<, <=, >, >=	விடச் சிறிது, விடச் சிறிது அல்லது நிகர், விடப் பெரிது, விடப்பெரிது அல்லது நிகர்	
==, !=	நிகரானது, நிகரில்லை	
&&	தருக்க AND	
||	தருக்க OR	
?:	நிபந்தனை செயற்குறி	
=	மதிப்பிருத்தல்	
+=, -=, *=, /=	குறுக்கு வழி மதிப்பிருத்து செயற்குறி	
,	காற்புள்ளி செயற்குறி	கீழ்

Chapter 9.indd 14 03/09/18 5:57 PM

உங்களுக்குத் தெரியுமா?

C++ -ல் ஒன்றிரண்டு செயற்குறிகள் வெவ்வேறு இடங்களில் வெவ்வேறு பொருள்படும்படி செயல்படுகிறது.
உதாரணமாக (*) எனும் செயற்குறி பெருக்கல் மற்றும் சுட்டு மாறி செயற்குறியாகவும் பயன்படுத்தப்படுகிறது.

9.3.5 நிறுத்தற்குறிகள் (Punctuators)

நிறுத்தற்குறிகள் என்பவை குறிப்பிட்ட பணியைச் செய்யும் குறியுருக்களாகும். C++ நிரலில் நிறுத்தற்குறிகள் வரம்புச்சுட்டி (Separator) அல்லது வரம்புக்குறியாகப் பயன்படுத்தப்படுகிறது. கீழ்வரும் நிறுத்தற்குறிகள் C++-ல் பயன்படுத்தப்படுவது போலவே C மற்றும் Java -நிரலாக்க மொழிகளிலும் பயன்படுத்தப்படுகிறது.

வரம்புச்சுட்டி	விளக்கம்	எடுத்துக்காட்டு

{ } நெளிவு அடைப்புக்குறி (Curly braces)	ஒரு நிரலின் தொகுதி, தொடக்கம் மற்றும் முடிவு நெளிவு அடைப்புக்குறிகளால் குறிக்கப்படுகிறது. ஒரு நிரலின் தொகுதி ஒன்று அல்லது ஒன்றிற்கு மேற்பட்ட செயல்பாட்டுக் கூற்றுகளைக் கொண்டிருந்தால் அது கலவை கூற்று (compound statement) எனப்படும்.	int main ( )
		{
		int x=10, y=20, sum;
		sum = x + y;
		cout << sum;
		}

( ) அடைப்புக்குறி (Parenthesis)	இந்த அடைப்புக்குறி, செயற்கூறு அழைப்பு (function call) மற்றும் செயற்கூறு அளபுருக்களைக் (function parameters) குறிக்கப் பயன்படுகிறது..	clrscr( ); int main ( )
[ ] சதுர அடைப்புக்குறி (Square brackets)	ஒரு பரிமாண மற்றும் பல பரிமாண அணிகளைக் குறிக்கப் பயன்படுகிறது.	int num[5]; char name[50];
, காற்புள்ளி (Comma)	இது ஒரு கோவையில் வரம்புச் சுட்டியாகப் பயன்படுத்தப்படுகிறது.	int x=10, y=20, sum;

; அரைப்புள்ளி (Semicolon)	ஒவ்வொரு c++ கூற்றுகளும் அரைப்புள்ளியில் கொண்டு முடிக்கப்படுகிறது.	int main ( )
		{
		int x=10, y=20, sum;
		sum = x + y;
		cout << sum;
		}

: முற்கால்புள்ளி (Colon)	இது ஒரு கூற்றின் அடையாளக் குறியீடு	private:

குறிப்புரைகள் (Comments)
//
/* */
	இவற்றில் உள்ளடங்கிய உரைத்தொகுதி குறிப்புரையாகக் கருதப்படும். நிரல் பெயர்ப்பி // என்ற குறியுடன் தொடங்கும் கூற்றை செயல்படுத்தாது.
	// ஒரு வரி குறிப்புரை
	/* */ பல வரி குறிப்புரை
		/* This is written
		By myself
		to learn CPP */
		int main ( )
		{
		int x=10, y=20, sum;
		// to sum x and y
		sum = x + y;
		cout << sum;
		}

Chapter 9.indd 15 03/09/18 5:57 PM

சுய மதிப்பீடு

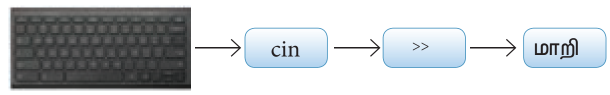
1. செயற்குறிகளின் பயன் யாது?
2. இரும செயற்குறி என்றால் என்ன? கணித இரும செயற்குறிகளுக்கு எடுத்துக்காட்டு தருக.
3. வகுமீதி (%) செயற்குறியின் செயல்பாடு என்ன?
4. 8.5 % 2 – விடை யாது?
5. R-ன் தொடக்க மதிப்பு 35 எனில் S=(R--)+(++R), என்ற கோவையில், S-யின் மதிப்பு என்ன?
6. தொடக்கத்தில் k=20 எனில் j=--k+2k என்ற கோவையின் விடை யாது?
7. j=22 மற்றும் p=3 எனில் p=p*++j என்பதன் விடை யாது?
8. i=8, j=10 மற்றும் k=8 எனில் பின்வரும் கோவைகள் மதிப்பு என்ன?
   (i) i<k (ii) i<j (iii) i>=k
   (iv) i==j (v) j!=k
9. பின்வரும் கோவைகளின் மதிப்பீட்டு வரிசை முறையை எழுது.
   (i) i+3>j-9 (ii) a+10<p-3+2q
10. பின்வரும் கூற்றுக்கு தருக்க கோவையை எழுதுக. மதிப்பெண் 75 எனில் தரம் A

9.4 உள்ளீட்டு-வெளியீட்டு செயற்குறிகள்

9.4.1 உள்ளீட்டுச்செயற்குறி (Input operator):

>> என்பது C++ ன் உள்ளீட்டு செயற்குறியாகும். இது விசைப்பலகையின் மூலம் மதிப்புகளைப் பெற்று வலப்பக்கம் உள்ள மாறியில் மதிப்பை இருத்துகிறது. எனவே, இது தரவு ஈர்ப்பு (extraction) அல்லது தரவு பெறும் (get from) செயற்குறி எனப்படும்.

இந்த செயற்குறிக்கு இரு செயலேற்பிகள் தேவைப்படுவதால், இது ஒரு இரும செயற்குறியாகும். விசைப் பலகையின் மூலம் பெறப்படும் உள்ளீட்டினை, முன்னதாகவே வரையறுக்கப்பட்ட cin என்னும் பொருளால் (சி-இன் என்று உச்சரிக்கப்படுகிறது) அடையாளம் காண்கிறது, இது முதல் செயலேற்பியாகும். இரண்டாவது செயலேற்பி ஒரு மாறியாக இருக்க வேண்டும்.



படம் 9.4 cin-ன் செயல்படும் முறை

ஒன்றுக்கு மேற்பட்ட தரவு மதிப்புகளை >> எனும் செயற்குறி மூலம் பெற்று உரிய மாறிகளில் இருத்தி வைக்க முடியும். இதற்கு செயற்குறித் தொடராக்கம் (cascading of operator) என்று பெயர்.

எடுத்துக்காட்டு:

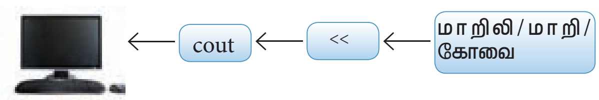
cin >> num;	தரவு ஈர்ப்புச் செயற்குறி, உள்ளீட்டுத் தாரைப் பொருளான cin-லிருந்து தரவினை ஈர்த்து, வலப்புறமுள்ள மாறி num-ல் இருத்துகிறது.
cin >>x >> y;	இது இரண்டு மதிப்புகளை ஈர்க்கும். cin முதல் மதிப்பை உடனடியாகப் பெற்று x என்னும் மாறியிலும் இரண்டாவது மதிப்பைப் பெற்று y என்னும் மாறியில் இருத்துகிறது.

9.4.2 வெளியீட்டு செயற்குறி (Output Operator):

<< என்பது C++ல் உள்ள வெளியீட்டு செயற்குறியாகும். வலதுப்பக்கம் உள்ள மாறியின் மதிப்புகளைப் பெற்று இடதுப்பக்கம் உள்ள முன் வரையறுக்கப்பட்டுள்ள பொருளான cout-க்கு அனுப்புகிறது. இது தரவு விடுப்பு (insertion) அல்லது தரவு தரும் (put-to) செயற்குறியாகும். இரு ஒரு இரும செயற்குறியாகும்.

Chapter 9.indd 16 03/09/18 5:57 PM

முன் வரையறுக்கப்பட்ட cout என்னும் பொருள் (சி-அவுட் என்று உச்சரிக்கப் படுகிறது) வெளியீட்டு சாதனமான திரையகத்தை அடையாளம் காண்கிறது. இதில் இரண்டாம் செயலேற்பி ஒரு மாறிலி, மாறி அல்லது கோவையாக இருக்கலாம்.



படம் 9.5 cout-ன் செயல்படும் முறை

ஒரே சமயத்தில் ஒன்றுக்கு மேற்பட்ட மாறி / மாறிலி / கோவையின் மதிப்புகளை << என்னும் செயற்குறி மூலம் திரையகத்திற்கு அனுப்பப்படுகிறது. இதற்கு செயற்குறித் தொடராக்கம் (cascading of operator) என்று பெயர்.

எடுத்துக்காட்டு:

cout << “Welcome”;	முன்வரையறுக்கப்பட்ட பொருளான cout, welcome என்னும் சரத்தை திரைக்கு அனுப்புகிறது..
cout << “The sum = “ << sum;	முதலில் “The sum=” என்னும் சரத்தையும், பிறகு sum என்னும் மாறியின் மதிப்பையும் திரைக்கு அனுப்புகிறது. பொதுவாக, cout எனும் பொருள் இரட்டை மேற்கோள்குறிகளுக்குள் தரப்படும் சர நிலையுறு அல்லது ஒற்றை மேற்கோள் குறிகளுக்குள் தரப்படும் குறியுரு மாறிலி (விடுபடு வரிசை தவிர) திரைக்கு அனுப்புகிறது.
cout <<“\n The Area: “ <<3.14*r*r;	cout – எனும் பொருள் இரட்டை மேற்கோள் குறிக்குள் உள்ள அனைத்தையும் திரைக்கு அனுப்பும் (\n தவிர). பிறகு கோவை 3.14*r*rயை மதிப்பீடு செய்து திரைக்கு அனுப்பும்.(\n - விடுபடு வரிசை புதிய வரியை உருவாக்கும்.).
cout << a + b ;	a மற்றும் b யின் கூட்டுத்தொகையை திரைக்கு அனுப்புகிறது.

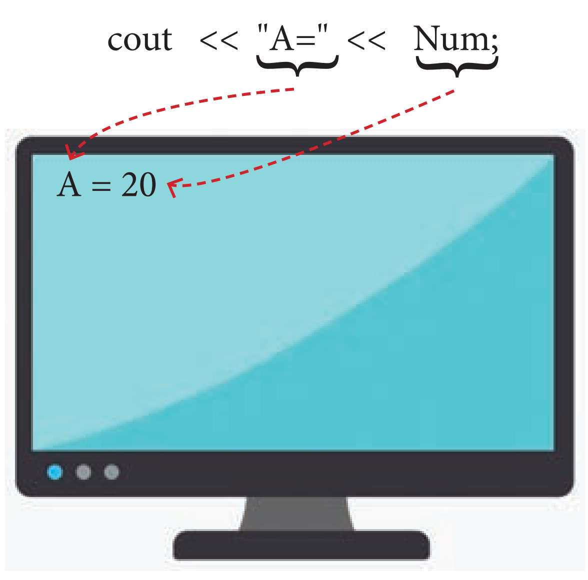
9.4.3. I/O செயற்குறிகள் தொடராக்கம் (Cascading of I/O operators)

ஒன்றுக்கும் மேற்பட்ட உள்ளீட்டு மற்றும் வெளியிட்டு செயற்குறிகள் >> மற்றும் << ஒரே கூற்றில் பயன்படுத்தப்பட்டால் அதற்கு I/O செயற்குறிகளின் தொடராக்கம் என்று பெயர்.

cout ன் தொடராக்கம் (Cascading cout):

```
int Num=20;
cout << “A=” << Num;
```

படம் 9.6 cout-ன் தொடராக்கத்தை விளக்குகிறது.



படம் 9.6 cout-ன் தொடராக்கம்

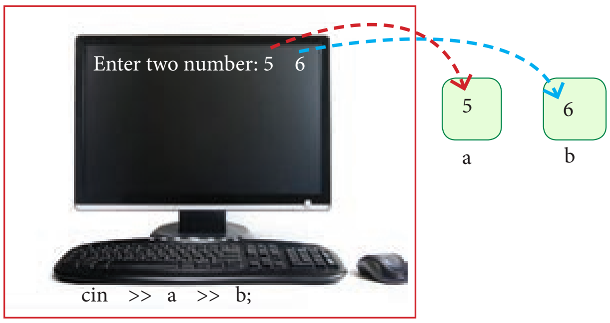
Chapter 9.indd 17 03/09/18 5:57 PM

cin ன் தொடராக்கம் (Cascading cin)

cout >> “Enter two number: ”;

cin >> a >> b;

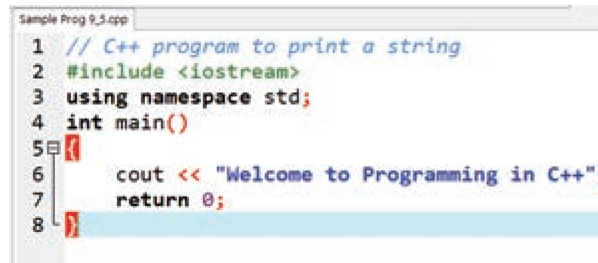
படம் 9.7 cin –ன் தொடராகத்தை விளக்குகிறது.



படம் 9.7 cin தொடராக்கம்

9.5 C++ -ல் முதல் எடுத்துக்காட்டு நிரல்

“Welcome to Programming in C++” எனும் சரத்தை அச்சிடும் நமது முதல் C++ நிரல் காண்போம்.



மேற்கண்ட நிரல் கீழ்காணும் வெளியீட்டைத் தருகிறது

Welcome to Programming in C++

இது அடிப்படை கூறுகளை உள்ளடக்கிய ஒரு எளிய C++ நிரல் ஆகும். இந்த கூறுகளைப் பற்றிப் பார்ப்போம்.

1	**// C++ program to print a string**
	இது ஒரு குறிப்புரை கூற்று. // என்ற குறியுடன் தொடங்கும் அனைத்து கூற்றுகளும் குறிப்புரைகளாகும். நிரலில் உள்ள குறிப்புரைகளை நிரல் பெயர்ப்பியானது புறக்கணித்து, அவற்றைச் செயல்படுத்தாது. ஒன்றிக்கு மேற்பட்ட குறிப்புரைகளை /* ……. */ என்ற குறிக்குள் தர வேண்டும்.
2	**# include <iostream>**
	அனைத்து C++ நிரல்களும் include கூற்றுடன் # குறியுடன்தொடங்கும். # என்பது ஒரு முன்செயலி நெறியுறுத்தும். இந்த கூற்றுகள் நிரல் தொகுப்புக்கு முன்னதாகவே செயல்பட தொடங்கும். #include <iostream> எனும் கூற்று, iostream என்னும் தலைப்புக் கோப்பினை நிரலில் சேர்த்துக் கொள்ளுமாறு நிரல் பெயர்ப்பிக்கு உணர்த்துகிறது. உள்ளீடு / வெளியீடு செயல்பாடுகளை பயன்படுத்தி கொள்ள வேண்டுமெனில் iostream என்னும் தலைப்பு கோப்பை ஒவ்வொரு C++ நிரலிலும் இணைக்க வேண்டும். சுருக்கமாக, iostream என்னும் தலைப்புக் கோப்பு, அதன் பொருள் உறுப்பினர்களான cin மற்றும் cout –டினை கொண்டுள்ளது. iostream என்னும் தலைப்புகோப்பைநிரலில்கொடுக்க தவறினால் cin மற்றும் cout–க்கானபிழைச்செய்தியை சுட்டிக்காட்டும் மற்றும் உள்ளீட்டைப் பெறவோ அல்லது வெளியீட்டை அனுப்பவோ முடியாது.

Chapter 9.indd 18 03/09/18 5:57 PM

3 using namespace std;

using namespace std; என்னும் வரியானது, standard namespace -சைப் பயன்படுத்துமாறு நிரல் பெயர்ப்பிக்கு கூறுகிறது. Namespace என்பது இனக்குழு, பொருள் மற்றும் மாறிகளுக்குத் தேவையான குறிப்பெயர்களின் தொகுப்பாகும். ஒரு பெரிய செயல்திட்டத்தில் ஏற்படும் பெயர் முரண்பாடுகளைத் தவிர்க்கும் வழிமுறைகளை namespace கொடுக்கிறது. ANSI C++ தரக்குழுவினால் புதியதாக அறிமுகப்படுத்தப்பட்டுள்ளது.

4 int main ( )

C / C++ நிரலானது செயற்கூறுகளின் தொகுப்பாகும். ஒவ்வொரு C++ நிரலும் main( ) செயற்கூறினைக் கட்டாயமாக்கப் பெற்றிருக்க வேண்டும். செயல்படுத்தப்படும் கூற்றுகள் main( ) செயற்கூறினுள் இருக்க வேண்டும்.

```
5 {
6     cout << "Welcome to Programming in C++";
7     return 0;
8 }
```

நெளிவு அடைப்புக் குறிக்குள் {} உள்ள (வரிசை எண் 5 முதல் 8 வரை) கூற்றுகள் செயல்படுத்தப்படும் கூற்றுகளாகும். இது தொகுதி குறிமுறை எனப்படும். வரி 6-ல் உள்ள cout என்ற கட்டளை Welcome to Programming in C++ என்ற சரத்தை திரைக்கு அனுப்புகிறது. ஒவ்வொரு செயல்படுத்தப்படும் கூற்றுகளும் அரைப்புள்ளியுடன் (;) முடிய வேண்டும். வரி 7ல் உள்ள return எனும் சிறப்புச் சொல் ஒரு செயற்கூற்றுக்கு குறிப்பிட்ட மதிப்பை திருப்பி அனுப்ப பயன்படுகிறது. இதில் main( ) செயற்கூறுக்கு 0 என்ற மதிப்பை திருப்பி அனுப்புகிறது.

9.6 C++ நிரலை இயக்குதல் (Execution of C++ program)

C++ நிரலை உருவாக்குவதற்கும், இயக்குவதற்கும் பின்வரும் நான்கு முக்கிய படிநிலைகளை கட்டாயமாக பின்பற்ற வேண்டும்..

(1) மூலக்குறிமுறையை உருவாக்குதல் (Creating Source code) செல்லத்தக்க C++ குறிமுறைகளை, C++ தொகுப்பானில் விதிமுறைகளை பின்பற்றி தட்டச்சு செய்து உருவாக்கப்படுவதாகும்.

(2) .cpp நீட்டிப்பு பெயருடன் மூலக்குறிமுறையை சேமித்தல் (Saving source code with extension .cpp)

மூலக்குறிமுறையை தட்டச்சு செய்து முடித்த பின் கண்டிப்பாக .cpp என்ற நீட்டிப்பு பெயருடன் சேமிக்க வேண்டும்.

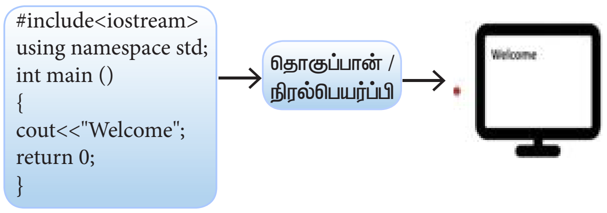
(3) தொகுத்தல் (Compilation)

நிரலை உருவாக்குவதில் இது ஒரு முக்கியமான படிநிலை ஆகும். தொகுத்தலில் முறையில் தொகுப்பான் நூலகக் கோப்புகளை மூல குறிமுறையுடன் தொடர்பு படுத்தி குறிமுறையின் ஒவ்வொரு வரியையும், சரிபார்க்கிறது. இதில் ஏதேனும் தவறோ அல்லது பிழையோ கண்டறியப்பட்டால் அதை திருத்துவதற்கு பயனருக்கு தெரிவிக்கிறது. எந்த பிழையும் இல்லை எனில் இது மூலக்குறிமுறையை கணினிக்கு புரிகின்ற இலக்கு கோப்பாக .obj என்ற நீட்டிப்புடன் மாற்றம் செய்கின்றது.

(4) இயக்குதல் (execution)

இது நிரலை உருவாக்குவதில் இறுதியான படிநிலை ஆகும். இந்த நிலையில், இலக்க கோப்பு. exe என்ற இயக்க கோப்பாக மாறுகிறது. நிரல், இயக்க கோப்பானதும் தனிச்சையாக இயங்கும். அதாவது, தொகுப்பான் அல்லது IDE யின் உதவியின்றி பயன்பாட்டை இயக்க முடியும்.

Chapter 9.indd 19 03/09/18 5:57 PM



படம் 9.8 இயக்குதல்

9.7 C++ உருவாக்கு சூழல் (C++ Development Environment)

C++ நிரல்களை உருவாக்க பல ஒருங்கிணைந்த உருவாக்க சூழல்கள் IDE உள்ளன. C++ எளிதாக நிரல்களை உருவாக்க, தொகுக்க மற்றும் இயக்க IDE மிகவும் உதவுகிறது. பல IDE - க்கள் திறந்த மூல (open source) பயன்பாடுகளாகும். அதாவது, இதனை இலவசமாக பெற முடியும்.

9.7.1 நன்கு அறியப்பட்ட IDEயுடனான C++ தொகுப்பான்கள்(Familiar C++ Compilers with IDE)

அட்டவணை 9.4 திறந்த மூல தொகுப்பான்கள்

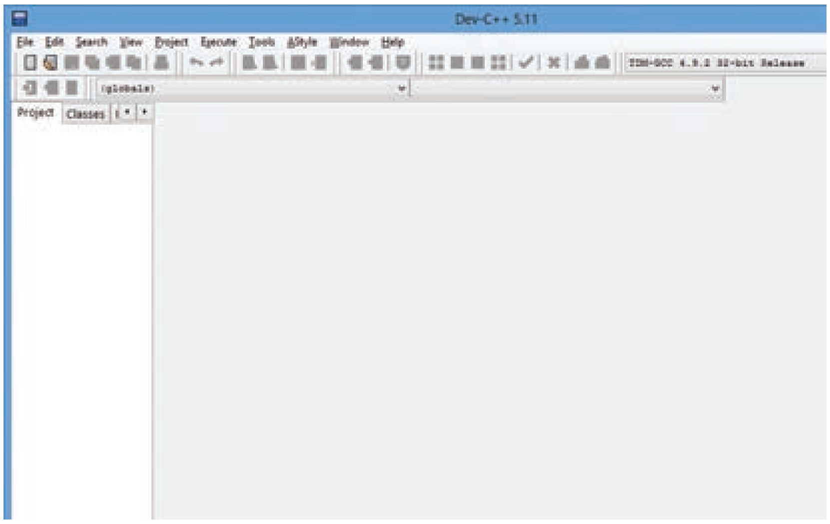
தொகுப்பான்	கிடைப்பு நிலை
Dev C++	திறந்த மூலம்
Geany	திறந்த மூலம்
Code::blocks	திறந்த மூலம்
Code Lite	திறந்த மூலம்
Net Beans	திறந்த மூலம்
Digital Mars	திறந்த மூலம்
Sky IDE	திறந்த மூலம்
Eclipse	திறந்த மூலம்

9.7.2 Dev C++ ல் வேலை செய்தல் (Working with Dev C++)

C++ நிரல்களை உருவாக்க பல IDE – க்கள் உள்ளன. ஆனால் இந்த பாடப்புத்தகத்தில் “Dev C++” தொகுப்பானை பயன்படுத்துகிறோம். இந்த புத்தகத்தில் கொடுக்கப்பட்டுள்ள நிரல்களும், செயல்முறை நுட்பங்கங்களும் “Dev C++” தொகுப்பானை தழுவியதாகும்.

Dev C++ ஒரு திறந்த மூல (Open Source) பல இயக்க அமைப்புகளிலும் இயக்கும் பணித்தளம் (Cross platform). லினக்ஸ் இயக்க அமைப்பில் பதிப்பு Dev C++ alpha பதிப்பை பயன்படுத்தலாம். இந்த பதிப்பு பகிர்ந்தளிக்கப்பட்ட GNU (General Public license) யுடன் கூடிய முழுமையான ஒருங்கிணைந்த உருவாக்க சூழல் ஆகும். இது Delphi யில் உருவாக்கப்பட்டுள்ளது. இதை http://www.bloodshed.net/dev/devcpp.html லிருந்து பதிவிறக்கம் செய்யலாம்.

1. Dev C++ ஐ நிறுவிய பிறகு, Dev C++ பணிக்குறியானது திரைமுகப்பில் தோன்றும். IDE- யை திறப்பதற்கு இரட்டை கிளிக் செய்யவும். Dev C++ IDE பின்வருமாறு தோன்றும்.

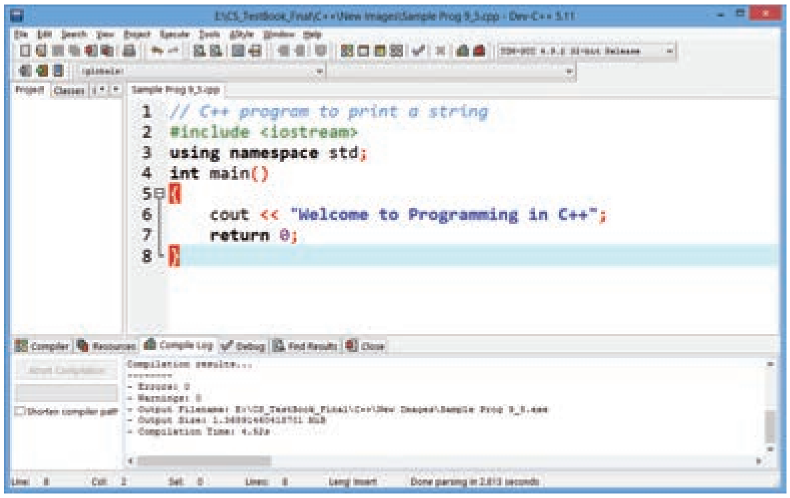


படம் 9.9 Dev C++ தொடக்க சாளரம்

Chapter 9.indd 20 03/09/18 5:57 PM

2. மூலக் கோப்பினை (source file) உருவாக்குவதற்கு **File →New →Source file** அல்லது **Ctrl + N** என்பதை தேர்ந்தெடுக்கவும்.

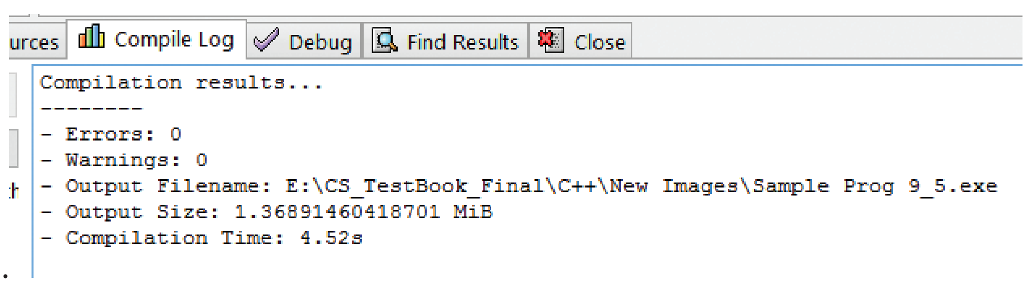
3. தோன்றுகின்ற திரையில் C++ நிரலை தட்டச்சு செய்து அதை சேமிக்க File → Save அல்லது **Ctrl + S** என்பதை பயன்படுத்த வேண்டும். இது தானமைவாக, cpp என்ற நீட்டிப்பை கோப்பின் பெயருடன் சேர்த்துக்கொள்ளும். கோப்புப் பெயருடன் .cpp என்பதை தட்டச்சு செய்யத் தேவையில்லை.



படம் 9.10 C++ நிரலுடன் கூடிய Dev C++ சாளரம்

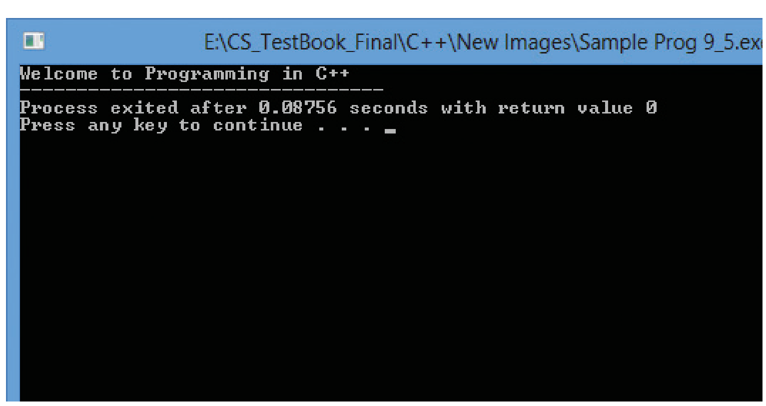
4. கோப்பினை சேமித்தப்பின், **Execute → Compile & Run** என்பதை கிளிக் செய்யவும் அல்லது F11 விசையை அழுத்தவும்.

நிரலில் ஏதேனும் பிழை இருந்தால் அது Compile log - ல் காண்பிக்கப்படும். நிரலில் பிழை ஏதும் இல்லை என்றால் பின்வரும் காட்சி தோன்றும்.



படம் 9.11 Dev C++ Compile Log

5. வெற்றிகரமான தொகுப்பிற்கு பிறகு வெளியீடு Output console பகுதியில் பின்வருமாறு தோன்றும்.



படம் 9.12 Dev C++ வெளியீட்டு சாளரம்.

Chapter 9.indd 21 03/09/18 5:57 PM

9.8 பிழைகளின் வகைகள் (Types of Errors)

பொதுவான பிழை வடிவங்கள் பின்வருமாறு:

பிழையின் வகைகள்	விளக்கம்
இலக்கணப்பிழை அல்லது தொடரியல் பிழை (Syntax Error)	• தொடரியல் அல்லது இலக்கணம் (Syntax) என்பது நிரல்களை உருவாக்குவதற்கு தேவையான இலக்கண விதிமுறைகளின் தொகுப்பாகும். ஒவ்வொரு நிரலாக்க மொழிக்கும் மூலக் குறிமுறையை உருவாக்குவதற்கு தனி தனி விதிமுறைகள் உள்ளன. • C++ ன் இலக்கண விதிமுறைகள் மீறப்படும் போது தொடரியல் பிழைகள் ஏற்படுகின்றன. • எ.கா. பின்வருமாறு தட்டச்சு செய்தால் C++ பிழையை ஏற்படுத்தும். cout << "Welcome to Programming in C++" C++ இலக்கண விதிமுறைகளின் படி அனைத்து இயக்க கூற்றுகளும் கண்டிப்பாக அரைப்புள்ளியுடன் (;) முற்றுப்பெற வேண்டும். ஆனால் இந்த கூற்று அரைப்புள்ளியுடன் (;)முடிக்கப்படவில்லை.
சொற்றொடர் பிழை (Semantic Error)	• ஒரு நிரலானது இலக்கண விதிமுறை படி சரியாக இருந்தும், தேவையான விடையை வழங்காமல் இருக்கலாம். ஏனெனில் மாறி / செயற்குறி / இயக்கப்படும் வரிசை போன்றவற்றில் ஏதேனும் தவறு இருப்பின், இந்த பிழையானது தோன்றும். இதன்படி, நிரலானது இலக்கண விதிமுறைப்படி சரியாக இருந்து தருக்க (logic) முறைப்படி தவறாக உள்ளது. ஆகையால் சொற்றொடர் பிழை தருக்க பிழை, என்றும் அழைக்கப்படும்.
இயக்க நேர பிழை (Run-time error)	• ஒரு நிரலை இயக்கும் போது, இயக்க நேரப்பிழை தோன்றலாம். காரணம், முறையில்லாத செயல்முறைகளால் இந்த பிழை ஏற்படும். • எடுத்துக்காட்டாக: இல்லாத ஒரு கோப்பை நிரலானது திறக்க முற்படும் போது இயக்க நேரப்பிழை ஏற்படுகிறது.

நினைவில் கொள்க

- C++ 1979ம் ஆண்டு AT & T பெல் ஆய்வுக் கூடத்தில் ஜேர்ன் ஸ்ட்ரௌஸ்ட்ரப் அவர்களால் உருவாக்கப்பட்டது.
- குறியுருத் தொகுதி என்பது ஒரு C++ நிரலை எழுதுவதற்கு அனுமதிக்கப்படும் எழுத்துகளின் தொகுப்பாகும்.
- C++ சிறப்புச் சொற்கள் நிரல் பெயர்ப்பிக்கு குறிப்பிட்ட அர்த்தத்தை தெரிவிக்கும் சொற்களாகும் .
- குறிப்பெயர்கள் என்பது C++ நிரலில் வெவ்வேறு பகுதிகளுக்கு கொடுக்கப்படும் பெயர்களாகும். இவை பயனரால் வரையறுக்கப்பட்ட, மாறிகள், செயற்கூறுகள், அணிகள், இனக்குழுக்கள் போன்றவை ஆகும்.
- ஒரு நிரலை நிறைவேற்றும் போது மதிப்புகள் மாறாத தரவுகள் நிலையுருக்கள் ஆகும். எனவே, நிலையுருக்கள் மாறிலிகள் என்றும் அழைக்கப்படுகிறது.
- C++ல் பல வகையான நிலையுருக்கள் உள்ளன. முழுஎண், மிதப்புப் புள்ளி எண் எழுத்துரு சரம்.
- செயற்குறிகள் கணித, தருக்க செயல்பாடுகளை செய்யப் பயன்படும் குறியீடுகளாகும்.
- C++ நிரலை உருவாக்குவதற்கு வரம்புக்குறிகளாக பயன்படுத்தப்படும் குறியீடுகள் நிறுத்தற்குறிகளாகும். இவை வரம்புச் சுட்டி என்றும் அழைக்கப்படுகிறது.
- தரவு ஈர்ப்பு(>>) மற்றும் தரவு விடுப்பு (<<) தரவை பெறவும், வெளியீட்டை அனுப்பவும் பயன்படுகிறது.

செய்முறை பயிற்சி

- பின்வரும் C++ நிரலை Dev C++ IDE- யில் தட்டச்சு செய்து இயக்குக. நிரல்பெயர்ப்பி / தொகுப்பான் ஏதேனும் பிழையை சுட்டிக்காட்டினால் அதை சரி செய்து, சரியான விடை வரும் வரை மீண்டும் மீண்டும் இயக்கவும்.

1. மூன்று பாட மதிப்பெண்களின் கூட்டுத்தொகையை கண்டறியும் C++ நிரல்

```
#include <iostream>
using namespace std;
int main()
{
        int m1, m2, m3, sum;
        cout << "\n Enter Mark 1: ";
        cin >> m1;
        cout << "\n Enter Mark 2: ";
        cin >> m2;
        cout << "\n Enter Mark 3: ";
        cin >> m3;
        sum = m1 + m2 + m3;
        cout << "\n The sum = " << sum;
}
```


Chapter 9.indd 23 03/09/18 5:57 PM

- கொடுக்கப்படும் மதிப்பெண்களின் சராசரியை கணக்கிடும் வகையில், மேலே கொடுக்கப்பட்டுள்ள நிரலில் வேண்டிய மாற்றங்களை செய்து இயக்குக.

2. வட்டத்தின் பரப்பளவை கண்டறியும் C++ நிரல்.

```
#include <iostream>
using namespace std;
int main()
{
        int radius;
        float area;
        cout << "\n Enter Radius: ";
        cin >> radius;
        area = 3.14 * radius * radius;
        cout << "\n The area of circle = " << area;
}
```

3. கீழே கொடுக்கப்பட்டுள்ள நிரலின் பிழைகளை கண்டறிக

```
Using namespace std;
int main( )
{
cout << “Enter a value ”;
cin << num1 >> num2
num+num2=sum;
cout >> “\n The Sum= ” >> sum;
```

4. கீழே கொடுக்கப்பட்டுள்ள நிரலில் எந்த வகையான பிழை உள்ளது என்று கண்டறிக:

```
#include <iostream>
using namespace std;
int main()
{
        int h=10; w=12;
        cout << "Area of rectangle " << h+w;
}
```


Chapter 9.indd 24 03/09/18 5:57 PM

மதிப்பீடு

பகுதி – அ

சரியான விடையை தேர்வு செய்யவும் :

1. C++ -யை உருவாக்கியவர் யார்?

(அ) சார்லஸ் பாபேஜ் (ஆ) ஜேர்ன் ஸ்ட்ரௌஸ்ட்ரப்

(இ) பில் கேட்ஸ் (ஈ) சுந்தர் பிச்சை

2. C ++ க்கு முதலன்முதலில் வைக்கப்பட்ட பெயர் என்ன?

(அ) சிபிபி (ஆ) மேம்பட்ட சி

(இ) இனக்குழுக்கள் உடன் சி (ஈ) சி உடன் இனக்குழுக்கள்

3. C ++ என பெயர் சூட்டியவர் யார்?

(அ)) ரிக் மாஸ்கிட்டி (ஆ) ரிக் பிஜர்னே

(இ) பில் கேட்ஸ் (ஈ) டென்னிஸ் ரிட்சி

4. ஒரு நிரலில் உள்ள மீச்சிறு தனித்த அலகு:

(அ) நிரல் (ஆ) நெறிமுறை

(இ) பாய்வுப்படம் (ஈ) வில்லைகள்

5. பின்வரும் செயற்குறிகளில் C ++ இன் தரவு ஈர்ப்பு செயற்குறி எது?

(அ) >> ஆ)<<

(இ)<> (ஈ) ^^

6. பின்வரும் வாக்கியங்களில் எது உண்மை இல்லை?

(அ) நிரல்பெயர்ப்பிக்கு மட்டுமே புரிகின்ற பொருள் கொண்ட காப்பு சொற்களுக்கு சிறப்பு சொற்கள் என்று பெயர்.

(ஆ) ஒதுக்கப்பட்ட சொற்கள் அல்லது முக்கிய சொற்களை குறிப்பெயராகப் பயன்படுத்தலாம்.

(இ) முழு எண் மாறிலி தசம புள்ளி இல்லாமல் குறைந்தபட்சம் ஒரு இலக்கத்தை கொண்டிருக்க வேண்டும்.

(ஈ) அடுக்கு மாறிலிகளின் வடிவம் இரண்டு பகுதிகளைக் கொண்டுள்ளது

7. கீழே கொடுக்கப்பட்டவைகளில் எது ஒரு சரநிலையுரு அல்ல?

(அ) 'A' (ஆ) 'Welcome' (இ) 1232 (ஈ)"1232"

8. உயர்நிலை மொழியில் எழுதப்படும் நிரல் எவ்வாறு அழைக்கப்படுகிறது ?.

(அ) இலக்கு குறிமுறை (ஆ) மூல குறிமுறை

(இ) இயங்கக்கூடிய குறிமுறை (ஈ) இவை அணைத்தும்

9. a=5, b=6; எனில் a& b யின் விடை என்ன?

(அ) 4 (ஆ) 5

(இ) 1 (ஈ) 0

Chapter 9.indd 25 03/09/18 5:57 PM

10. தொகுப்பு நேர (Compile time) செயற்குறி எவ்வாறு அழைக்கப்படுகிறது ?.

(அ) sizeof (ஆ)pointer

(இ) virtual (ஈ) this

பகுதி – ஆ

குறு வினாக்கள் (2 மதிப்பெண்கள்) :

1. வில்லைகள் என்றால் என்ன? C++ -ல் உள்ள வில்லைகளை கூறுக.

2. சிறப்புச் சொற்கள் என்றால் என்ன? சிறப்புச் சொற்களை குறிப்பெயர்களாக பயன்படுத்தலாமா?

3. பின்வரும் மாறிலிகள் எந்த வகை மாறிலிகள் ஆகும்?

(i) 39 (ii) 032 (iii) 0XCAFE (iv) 04.14

4. கீழ்க்கண்ட மெய்யான மாறிலிகளை அடுக்குகுறி வடிவில் எழுதுங்கள்:

(i) 23.197 (ii) 7.214 (iii) 0.00005 (iv) 0.319

5. n=10; எனில், n>>2; ன் விடை என்ன ?

6. பொருத்துக:

A	B
(a) வகுமீதி	(1) வில்லைகள்
(b) வரம்புச்சுட்டி	(2) வகுத்தலின் மீதி
(c) தரவு ஈர்ப்பு	(3) நிருத்தக்குறிகள்
(d) மொழித் தொகுதி	(4) தரவு பெறும்

பகுதி – இ

சிறு வினாக்கள் (3 மதிப்பெண்கள்) :

1. சிறப்புச் சொற்கள் (keywords) மற்றும் குறிப்பெயர்கள் (identifiers) –க்கு இடையே உள்ள வேறுபாடுகளை விவரி?

2. C++ ஒரு எழுத்து வடிவ உணர்த்தியா? எழுத்து வடிவ உணர்த்தி (case sensitive) என்பதன் பொருள் என்ன?

3. “=” மற்றும் “==” வேறுபடுத்துக.

4. a=10, b=15; எனில் a^b – யின் மதிப்பு என்ன?

5. தொடரியல் பிழை (Syntax error) மற்றும் இயக்க நேர பிழை (Run time error) இவற்றிக்கு இடையே உள்ள வேறுபாடுகளை எழுதுக.

Chapter 9.indd 26 03/09/18 5:57 PM

6. தொடரியல் பிழை (Syntax error) மற்றும் தர்க்கரீதியான பிழை (Logical error) இவற்றிக்கு இடையே உள்ள வேறுபாடுகளை கூறுக.
7. தலைப்புக் கோப்பின் பயன் யாது?
8. main செயற்கூற்றின் சிறப்பு யாது.?
9. உள்ளடக்க தொகுப்பானின் (include compiler directive) இரண்டு பயன்பாடுகளை எழுதுக.
10. மெய் மாறிலிகளில் (real constants) பின்வருவனவற்றை எழுதுக.

(i) 15.223 (ii) 211.05 (iii) 0.00025

பகுதி – ஈ

பெரு வினாக்கள் (5 மதிப்பெண்கள்) :

1. C++ -ல் பயன்படுத்தப்படும் இருநிலை செயற்குறிகளை பற்றி எழுதுக.
2. பிழைகளின் வகைகள் யாவை?
3. a=15, b=20; எனில் கீழ்காணும் செயல்பாட்டிற்கான விடை யாது.?

(a) a&b (b) a|b (c) a^b (d) a>>3 (e) (~b)

தரவினங்கள், மாறிகள் மற்றும் கோவைகள்

9.10 அறிமுகம்

அனைத்து நிரலாக்க மொழிகளிலும் இரண்டு அடிப்படை உறுப்புகள் உள்ளன அவை முறையே தரவு இனங்கள் மற்றும் மாறிகள். இவை மிகத் தொடக்க நிலை நிரலை எழுதுவதற்கு கூட தேவைப்படும். மிக முக்கிய உறுப்புகளாகும். தரவுகளை கையாள்வதற்காக முன்னதாகவே வரையறுக்கப்பட்ட தரவு இனங்களின் தொகுப்பை C++ வழங்குகிறது. இவை அடிப்படை அல்லது உள்ளிணைந்த தரவு இனங்கள் என்று அழைக்கப்படுகிறது. உள்ளிணைந்த தரவினங்கள் நீங்கலாக, நிரலரே சில தரவினங்களை உருவாக்க முடியும். அவை பயனர் வரையறுக்கும் தரவினம் என்று அழைக்கப்படுகிறது. இந்த பாடப்பகுதியில், உள்ளிணைந்த தரவினங்கள் பற்றி கற்கலாம்.

9.11 தரவு இனங்களின் கருத்தாக்கம்

பின்வரும் எடுத்துக்காட்டை உற்றுநோக்கவும்

Name = Ram

Age = 15

Average_Mark= 85.6

மேலே கொடுக்கப்பட்டட எடுத்துக்காட்டில் Name, Age, Average_Mark என்பன புலங்கள் ஆகும். இதில் இருத்தப்பட்டுள்ள Ram,15,85.6 என்பவை அப்புலத்தின் மதிப்புகளாகும.

செயல்முறை மொழியில் புலங்கள் மாறிகள் என்றும் மற்றும் மதிப்புகள் தரவுகள் என்றும் அழைக்கப்படுகிறது. மேலே கொடுக்கப்பட்டுள்ள ஒவ்வொரு தரவும் வேறுபட்டதாகும். அதாவது "Ram" என்பது எழுத்துக்களின் வரிசை மற்ற இரண்டு தரவுகளும் எண்களாகும். முதல் மதிப்பு ஒரு முழு எண்ணாகவும், இரண்டாம் மதிப்பு ஒரு தசம எண்ணாகவும் உள்ளது. நடப்புலக பயன்பாட்டில் மிகவும் வேறுபட்ட தரவு இனங்களை நாம் அன்றாடம் கையாள்கிறோம். தரவுகளின் தன்மை அல்லது வகைகள் வேறுபடுகிறது. எடுத்துக்காட்டாக தூரம் (உன் வீட்டிலிருந்து பள்ளிக்கு) நுழைவுச் சீட்டின் விலை, பேனாவின் விலை, மதிப்பெண்கள், வெப்பநிலை மற்றும் பல.

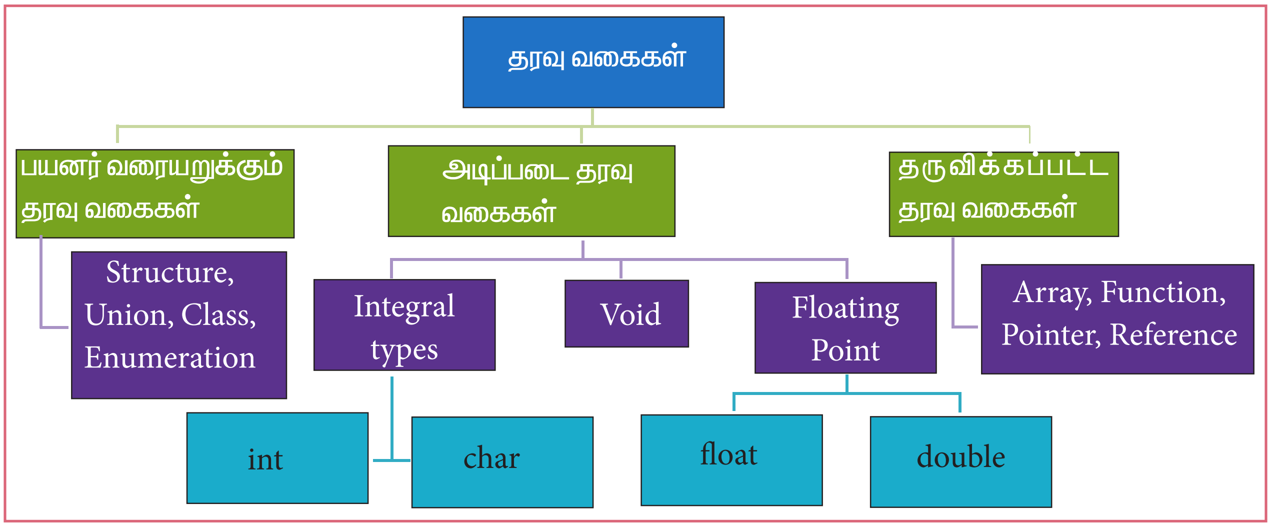
9.12 C++ தரவு இனங்கள்

C++ல் தரவு இனங்கள் மூன்று வகைகளாக பிரிக்கப்படுகிறது. அவை,

(1) அடிப்படை தரவு இனங்கள் (Fundamental data types)

(2) பயனர் வரையறுக்கும் தரவு இனங்கள் (User-defined data types and)

(3) தருவிக்கப்பட்ட தரவு இனங்கள் (Derived data types).



படம் 9.13 C++ ன் தரவினங்கள்

இந்த பாடப் பகுதியில் அடிப்படை தரவு இனங்களைப் பற்றி அறிந்து கொள்வோம். தரவு இனங்களின் வேலை செய்யும் விதத்தை பற்றி அறிந்து கொள்ள, மாறிகளைப் (Variable) பற்றி அறிந்து கொள்வது அவசியமாகும். மாறிகள் என்பவை குறிப்பிட்ட தரவு இனங்களின் மதிப்புகளை தாங்கியுள்ள, பெயரிடப்பட்ட நினைவாக இடங்களை குறிக்கிறது. C++ -ல் மாறிகளை பயன்படுத்துவதற்கு முன்னரே அவற்றை அதன் தரவினங்களோடு கண்டிப்பாக அறிவிக்கவேண்டும். அதாவது, எந்த வகையான தரவுகளை மாறியில் இருத்தப்போகிறோம் என்பதை தெளிவாக குறிப்பிட வேண்டும்.

மாறிகளை அறிவிப்பதற்கான கட்டளை அமைப்பு

< தரவினம் > < மாறியின் பெயர் >;

எடுத்துக்காட்டாக:

int num1;

ஒரே இனத்தை சார்ந்த ஒன்றுக்கு மேற்பட்ட மாறிகளை ஒரே அறிவிப்புக் கூற்றில் குறிப்பிட முடியும். அவை, காற்புள்ளியால்(,) பிரிக்கப்பட்டிருக்க வேண்டும்.

எடுத்துக்காட்டாக:

int num1, num2, sum;

மேலே கொடுக்கப்பட்ட கூற்றின் படி மதிப்பெண்களை சேமிக்க முதலில் மாறியை அதன் தரவினத்தோடு அறிக்க வேண்டும். தகுந்த தரவினத்தை தேர்ந்தெடுப்பதற்கு அறிவும், அனுபவமும்தேவைசாதாரணமாக,மதிப்பெண்கள் முழு எண்களாகவே இருக்கும். எனவே, கணிப்பொறி மதிப்பெண்ணை சேமிக்கும் மாறி முழு எண் தரவினமாக இருக்க வேண்டும்.

எடுத்துக்காட்டு:

int comp_science_marks;

இப்பொழுது comp_science_marks என்ற ஒரு மாறியில் மதிப்பெண் சேமிப்பதற்கு தயாராக உள்ளது.

Chapter 9.indd 28 03/09/18 5:57 PM

மாறிகளை பற்றி மேலும் பின்வரும் பகுதிகளில் தெரிந்து கொள்ளலாம்.

9.12.1 அடிப்படைத் தரவினங்கள் அறிமுகம்:

அடிப்படைத் தரவினங்கள்(atomic) C++ல். முன்னதாகவே வரையறுக்கப்பட்ட தரவினங்கள் ஆகும். C++ ல் ஐந்து அடிப்படை தரவினங்கள் உள்ளன. **char, int, float, double** மற்றும் **void.** பொதுவாக இவை தரவினங்களை அறிவிப்பதற்கான சிறப்பு சொற்கள் ஆகும்.

(i) முழுஎண் தரவினம் (int):

இது ஒரு தசமம் இல்லாத முழு எண்களாகும். முழு எண்கள் நேர்மறை அல்லது எதிர்மறை எண்களாக இருக்கலாம். இந்த தரவினம் முழு எண்களை மட்டுமே பெற்று அதையே திருப்புகிறது. முழு எண் (int) மாறிகள் முழு எண்களை மட்டுமே சேமிக்க அனுமதிக்கும். முழு எண் மாறியில் தசம மதிப்பை சேமிக்க முற்பட்டால் அந்த தசம மதிப்பினுடைய முழு எண் பகுதியை மட்டுமே எடுத்துக் கொள்ளும்.

எடுத்துக்காட்டு 9.1 இரண்டு முழு எண்களை பெற்று அதனுடைய கூட்டுத் தொகையை காண்பிக்கும் C++ நிரல்

```
#include <iostream>
using namespace std;
int main()
{
        int num1, num2, sum;
                        // variables num1, num2, and sum are declared as integers
        cout << "\n Enter Number 1: ";
        cin >> num1;
        cout << "\n Enter Number 2: ";
        cin >> num2;
        sum = num1 + num2;
        cout << "\n The sum of " << num1 << " and " << num2 << " is " << sum;
}
```

மேலே குறிப்பிடப்பட்டுள்ள நிரலில் அறிவிக்கப்பட்டுள்ள மூன்று மாறிகளான **num1,num2** மற்றும் **sum** ஆகிய மாறிகள் முழு எண்ணாகவே அறிவிக்கப்பட்டுள்ளது. எனவே, மூன்று வெவ்வேறு முழு எண் மதிப்புகளே இந்த மூன்று மாறிகளிலும் சேமிக்கப்படும். நிரல் இயக்கப்படும் போது பயனரால் கொடுக்கப்படுகின்ற மதிப்புகளை சேமிக்க num1, num2 என்ற மாறிகள் பயன்படுகின்றன. Sum என்ற மாறியானது செயல்படுத்தப்பட்ட மதிப்பை (விடை) சேமிக்கப் பயன்படுகிறது.

மேலே கொடுக்கப்பட்டுள்ள நிரல் இயங்கும் போது, வரிசை எண் 7 மற்றும் 9ல் குறிப்பிடப்பட்டுள்ள கூற்றுகள், num1மற்றும் num2க்கான மதிப்புகளை உள்ளீடாக கொடுக்கும் படி கேட்கும்.

வரிசை எண் 8 மற்றும் 10ல் பயனர் உள்ளீடாக தரும் மதிப்புகளை cin பெற்று முறையே மாறிகள் num1 மற்றும் num2ல் சேமிக்கும்.

வரிசை எண் 11, num1மற்றும் num2 மாறிலிகளில் உள்ள மதிப்புகளை கூட்டி, விடை மதிப்பை மாறி Sum-ல் சேமிக்கும்.

Chapter 9.indd 29 03/09/18 5:57 PM

(2) குறியுரு தரவினம் (char data type)

குருயுரு தரவினம் ஏற்றுக்கொள்ள கூடிய அனைத்து ASCII குறியுருக்களை பெற்று திருப்புகிறது. இது முழு எண் வகையாக கருதப்படுகிறது. ஏனெனில், அனைத்து குறியுருக்களும் நினைவகத்தில் அதனுடைய தொடர்புடைய ASCII குறியீடுகளாகவே குறிக்கப்படுகின்றன char என குறிப்பெயர் குறிக்கப்பட்டால் C++, குறியுரு அல்லது முழு எண் மதிப்பை சேமித்துக் கொள்ள அனுமதிக்கிறது.

எடுத்துக்காட்டு 9.2 ஏதேனும் ஒரு குறியுருவைப் பெற்று அதனுடைய அடுத்த குறியுருவை தெரிவிக்கும் நிரல்.

```
#include <iostream>
using namespace std;
int main()
{
        char ch;
        cout << "\n Enter a character: ";
        cin >> ch;
        ch = ch + 1;
cout << "\n The Next character: " << ch;
}
```

வெளியீடு:

```
        Enter a character: A
        The Next character: B
```

மேலே குறிப்பிடப்பட்ட நிரலில் **ch** என்ற மாறி**,** char தரவினமாக அறிவிக்கப்பட்டுள்ளது. பயனர் ஒரு குறியுருவை தரும் போது அது **ch** என்ற மாறியில் முழு எண் மதிப்பாக சேமிக்கப்படுகிறது. (அதாவது , சமமான ASCII குறியீடு)

ch = ch + 1;

இந்த கூற்றில் ch ன் மதிப்பு ஒன்று அதிகரிக்கப்பட்டு புதிய மதிப்பானது அதே ch மாறியில் இருத்தப்படுகிறது. (கணக்கீட்டுச் செயல்பாடானது எழுத்துக்களின் மீது அல்லாமல் எண்களில் மீது மட்டுமே செயல்படுகிறது).

முழு எண் மற்றும் குறியுரு தரவினங்கள் ஒன்றாக செயல்படும் விதத்தை விளக்கும் மற்றொரு நிரல்.

எடுத்துக்காட்டு நிரல் 9.3 ஒரு ASCII மதிப்பை பெற்று அதற்குரிய குறியுருவை காண்பிக்கும் C++ நிரல்

```
#include <iostream>
using namespace std;
int main ()
{
        int n;
        char ch;
        cout << "\n Enter an ASCII code (0 to 255): ";
        cin >> n;
        ch = n;
        cout << "\n Equivalent Character: " << ch;
}
```

வெளியீடு:

```
Enter an ASCII code (0 to 255): 100
Equivalent Character: d
```

மேலே உள்ள நிரலில் n என்ற மாறி முழு எண் int வகையாகவும், ch குறியுரு char வகையாகவும் அறிவிக்கப்பட்டுள்ளது. நிரல் இயக்கத்தின் போது ASCII மதிப்பை கொடுப்பதற்கு பயனுருக்கு நினைவூட்டுகிறது. பயனர் ASCII மதிப்பை முழு எண்ணாக தரும் பொழுது அது n என்ற மாறியில் சேமிக்கப்படுகிறது. ch=n என்ற கூற்றில் nன் மதிப்பு chல் இருத்தப்படுகிறது. ch என்பது குறியுரு வகை மாறி என்பதை நினைவில் கொள்க. எடுத்துக்காட்டாக, பயனர் “100” என்ற உள்ளீடை தரும் பொழுது முதலில் 100 என்ற மதிப்பு n என்ற மாறியில் சேமிக்கப்படுகிறது. அடுத்த கூற்றில் n ன் மதிப்பு அதாவது 100 ch ல் இருத்தப்படுகிறது. ch என்பது ஒரு குறியுரு வகை ஆகையால் அந்த எண்ணுக்குரிய மதிப்பு விடையாக வருகிறது.

(3) மதிப்புப் புள்ளி (float data type)

ஒரு மாறி float என குறிக்கப்பட்டால் தசம புள்ளி மதிப்புகளை மட்டும் மாறியில் சேமிக்கப்படும். தசம தரவினங்களில் பயன்படுத்துவதில் உள்ள இரண்டு சிறப்பு அம்சங்கள்

(1) முழு எண்களுக்கு இடையே உள்ள மதிப்பை குறிக்கிறது.

(2) மதிப்புகளின் மிக அதிகமான பரப்பை குறிக்கிறது. (greater range of values)

மிதப்பு புள்ளி அல்லது தசம செயற்பாடுகள் முழு எண் செயற்பாடுகளுடன் ஒப்பிடும் போது இயக்குதலுக்கு அதிக நேரத்தை எடுத்துக்கொள்ளும். அதாவது மிதப்பு புள்ளி செயற்பாடுகள் முழு எண் செயற்பாடுகளை விட மெதுவாக செயல்படும். இதுவே இதன் குறைபாடு ஆகும்.

Chapter 9.indd 31 03/09/18 5:57 PM

எடுத்துக்காட்டு 9.4 ஒரு வட்டத்தின் பரப்பளவைக் காண உதவும் C++ நிரல்

```
#include <iostream>
using namespace std;
int main()
{
        float r, area;
        cout << "\n Enter Radius: ";
        cin >> r;
        area = 3.14 * r * r;
        cout << "\n The Area of the circle is " << area;
}
```

வெளியீடு:

```
Enter Radius: 6.5
The Area of the circle is 132.665
```

மேலே கூறப்பட்ட எடுத்துக்காட்டில் 'r' மற்றும் 'area' என்ற இரண்டு மாறிகளும் மிதப்புப் புள்ளியாக ஒரே கூற்றில் அறிவிக்கப்பட்டுள்ளது. 'r' என்ற மாறி பயனரால் தரப்படுகின்ற ஆரத்தின் மதிப்பை எடுத்துக் கொள்கிறது. 'area' என்ற மாறி வாய்ப்பாட்டின் மூலம் பெறப்படும் வட்டத்தின் பரப்பளவைஎடுத்துக்கொள்ளும். எடுத்துக்காட்டாக பயனர் ஆரத்தின் மதிப்பை 6.5 என உள்ளீடு செய்தால்,

area = 3.14 *r*r;

'r' என்ற மாறியின் மதிப்பை இருமுறை பெருக்கி பிறகு 3.14 (pi- ன் மதிப்பு) உடனும் பெருக்கி விடை 132.665 யை மாறி areaல் சேமிக்கிறது.

இதில் 'r' மற்றும் 'area' என்பது ஒரே இன மாறியாகும் எனவே இந்த இரண்டு மாறிகளும் ஒரே கூற்றில் அறிவிக்கப்பட்டுள்ளது பல ஒரே இன மாறிகளை வெவ்வேறு கூற்றில் அறிவிப்பதற்கு பதிலாக ஒரே கூற்றில் அறிவிப்பதனால் செயல்பாட்டு நேரம் குறைகிறது.

(4) இரட்டை தரவினம் (Double data type)

இது துல்லியமான இரட்டை மிதப்புப் புள்ளி எண்களாகும். (துல்லியம் என்பது தசம புள்ளிக்கு பிறகு வரும் இலக்கங்களை குறிக்கிறது) இரட்டை மிதப்புப்புள்ளி எண்கள் சாதரண மிதப்புப் புள்ளி எண்களையும் கையாளும். ஆனால், இது மிதப்புப் புள்ளி தரவினத்தை விட இரண்டு மடங்கு இடத்தை எடுத்துக் கொள்ளும்.

(5) void data type:

void என்பதன் பொதுவான பொருள் வெற்றிடம். C++ void தரவின மதிப்புகள் ஒரு வெற்று தொகுப்பை குறிக்கிறது. இது எந்த மதிப்பையும் திருப்பி அனுப்பாது.

Chapter 9.indd 32 03/09/18 5:57 PM

சுய மதிப்பீடு

1. அடிப்படை தரவு வகை என்றால் என்ன?
2. char என்ற தரவினம் குறியுருக்களைக் குறிக்க பயன்படுகிறது. ஆனால் ஏன் அது முழு எண் தரவினமாக பெரும்பாலும் வரையறுக்கப்படுகிறது?
3. முழு எண்களை விட மிதப்புப் புள்ளி எண்கள் எத்த வகையில் சிறப்பானவை?
4. double என்பது மற்றொரு மிதப்பு புள்ளி தரவு வகை பிறகு ஏன் அதை தனி தரவினமாக பயன்படுத்துகிறோம்?
5. void தரவு இனத்தின் பயன் யாது?

9.12.2 அடிப்படை தரவினங்களின் நினைவக உருவமைத்தல் (Memory representation of Fundamental Data types)

நினைவகத்தில் தேவையான இடத்தை ஒதுக்குவதே மாறிகளை அறிவிப்பதற்கு மிக முக்கிய காரணம் ஆகும். நிரல் தேக்க கோட்பாட்டின்படி செயற்பாட்டிற்கு முன்னரே அனைத்து தரவுகளும் முதன்மை நினைவகத்தில் சேமிக்கப்பட வேண்டும். எனவே தொகுப்பானின் தரத்தின்படி ஒவ்வொரு தரவிற்கும் குறிப்பிட்ட நினைவக இடத்தை C++ தொகுப்பான் ஒதுக்குகிறது.

பின்வரும் அட்டவணை 9.5 அனைத்து அடிப்படைத் தரவினங்களுக்கும் எவ்வளவு இடம் ஒதுக்கப்படுகிறது. என்பதை காண்பிக்கிறது. கணிப்பொறி நினைவகத்தில் அனைத்து தரவுகளும் இரும எண்களாகவே சேமிக்கப்படும் என்பதை நினைவில் கொள்க.(அலகு 1 பாடம் 2 காண்க)

அட்டவணை 9. 5 அடிப்படை தரவினங்களின் நினைவக ஒதுக்கீடு

தரவினம்	நினைவகத்தின் இருப்பிடம்		மதிப்புகளின் பரப்பு
	பைட் -ல்	பிட் -ல்	
char	1 byte	8 bits	-127 to 128
int	2 bytes	16 bits	-32,768 to 32,767
float	4 bytes	32 bits	3.4×10^{-38} to 3.4×10^{38} -1
double	8 bytes	64 bits	1.7×10^{-308} to 1.7×10^{308}-1

9.12.3 தரவினங்களின் பண்புணர்த்திகள் (Data type modifiers)

மதிப்பிலி தரவினத்தை தவிர மற்ற அடிப்படை தரவினங்களின் சேமிக்கும் அளவை மாற்றி அமைக்க பண்புணர்த்திகள்(modifiers) பயன்படுகின்றன. சாதாரணமாகவே, நினைவகத்தில் தரவுகளை சேமிக்க அனைத்து அடிப்படை தரவினங்களும் ஒரு குறிப்பிட்ட அளவு பரப்பிலான மதிப்பை (Range of values) எடுத்துக்கொள்கிறது. எடுத்துக்காட்டாக, முழு எண் தரவினம் (integer) இரண்டும் பைட் தரவை மட்டுமே சேமிக்கும். உண்மையில், சில முழு எண் தரவுகள் மிகவும் நீளமானதாக மற்றும் அதிக நினைவக இடம் தேவைப்படுவதாகவோ இருக்கலாம். இந்த நிலையில், நினைவக அளவை கண்டிப்பாக மாற்றி அமைக்க வேண்டும். அடிப்படை தரவினங்களின் நினைவக ஒதுக்கீட்டை மாற்றி அமைக்க (விவரிக்க அல்லது சுருக்க) பண்புணர்த்திகள்பயன்படுகின்றன.இவைகள்தகுதியாக்கிகள்(Qualifiers)எனறும்அழைக்கப்படுகின்றன.

C++ ல் நான்கு பண்புணர்த்திகள்பயன்படுத்தப்படுகிறது, அவை

(1) signed (2) unsigned (3) long (4) short

இந்த நான்கு பண்புணர்த்திகளையும் எல்லா அடிப்படை தரவினங்களுடன் பயன்படுத்தலாம் பின்வரும் அட்டவணை 9.6 அனைத்து தரவினங்களின் நினைவக ஒதுக்கீட்டை பண்புணர்த்தியுடனும் அது இல்லாமலும் காண்பிக்கிறது..

அட்டவணை 9.6 குறியுரு தரவினத்தின் நினைவக வகைகள்

தரவினம்		நினைவக இருப்பிடம்		மதிப்புகளின் பரப்பு
		பைட் -ல்	பிட்-ல்	
short	short என்பது **short int** குறிக்கிறது	2 bytes	16 bits	-32,768 to 32,767
unsigned short	ஒரு முழு எண் எதிர்மறை குறியீடு இல்லாமல் இருப்பது	2 bytes	16 bits	0 to 65535
signed short	ஒரு முழு எண் எதிர்மறை குறியீடுடன் இருப்பது	4 bytes	32 bits	-32,768 to 32,767
short மற்றும் signed short இரண்டுமே சமமானது				
int	ஒரு முழு எண் என்பது குறியீடுடன் அல்லது குறியீடு இல்லாமல் இருக்கலாம்	2 bytes	16 bits	-32,768 to 32,767
unsigned int	ஒரு முழு எண் எதிர்மறை குறியீடு இல்லாமல் இருப்பது	2 bytes	16 bits	0 to 65535
signed int	ஒரு முழு எண் எதிர்மறை குறியீடுடன் இருப்பது	2 bytes	16 bits	-32,768 to 32,767
short மற்றும் int இரண்டுமே சமமானது				
long	**long** என்பது **long int** குறிக்கிறது	4 bytes	32 bits	-2147483648 to 2147483647
unsigned long	எந்த குறியீடும் இல்லாமல் இரட்டை இடைவெளி கொண்ட ஒரு முழு எண்	4 bytes	32 bits	0 to 4,294,967,295
signed long	குறியுடன் கூடிய இரட்டை இடைவெளி கொண்ட ஒரு முழு எண்	4 bytes	32 bits	-2147483648 to 2147483647

மேற்கண்ட அட்டவணையின் படி முழு எண் தரவினம் (integer) இரண்டு பைட் தரவுகளை மட்டும் எடுத்துக்கொள்கிறது. ஆனால் long int தரவினம் இந்த அளவை விட இரண்டு மடங்கு அதிகமான (4 பைட்டுகள்) தரவை எடுத்துக் கொள்கிறது. long in அதிக எண்களை சேமித்துக் கொள்கிறது. (long என்பது பண்புணர்த்தி) (int என்பது அடிப்படை தரவினம்)

அட்டவணை 9.7 குறியுரு தரவினத்தின் நினைவக ஒதுக்கீடு

தரவினம்	நினைவக இருப்பிடம்		மதிப்புகளின் பரப்பு
	பைட் -ல்	பிட்-ல்	
char	1 byte	8 bits	-128 to 127
unsigned char	1 byte	8 bits	0 to 255
signed char	1 byte	8 bits	-128 to 127

அட்டவணை 9.8 மதிப்புப் புள்ளி தரவினத்தின் நினைவக ஒதுக்கீடு

தரவினம்	நினைவக இருப்பிடம்		மதிப்புகளின் பரப்பு
	பைட் -ல்	பிட்-ல்	
float	4 bytes	32 bits	3.4×10^{-38} to $3.4\times10^{38}-1$
double	8 bytes	64 bits	1.7×10^{-308} to $1.7 \times 10^{308} - 1$
long double	10 bytes	80 bits	3.4×10^{-4932} to $1.1 \times 10^{4932} - 1$

C++ தொகுப்பானை பொறுத்து நினைவக ஒதுக்கீடு வேறுபடும். மேலே கொடுக்கப்பட்டுள்ள அட்டவணையில் உள்ள மதிப்புகள் Turbo C++ தொகுப்பானை அடிப்படையாக கொண்டது. int மற்றும் long double வகைக்கு Dev C++ல் அதிக இடத்தை கொடுக்கிறது.

பின்வரும் அட்டவணை 9.9 Turbo C++ மற்றும் DevC++ல் நினைவக ஒதுக்கீட்டிற்கான வேறுபாட்டை காட்டுகிறது.

அட்டவணை 9.9 Turbo C++ மற்றும் Dev C++ தொகுப்பான்களின் நினைவக ஒதுக்கீடு

தரவினம்	பைட்டுகளில் நினைவக அளவு	
	Turbo C++	Dev C++
short	2	2
unsigned short	2	2
signed short	2	2
int	2	4
unsigned int	2	4
signed int	2	4
long	4	4
unsigned long	4	4
signed long	4	4
char	1	1
unsigned char	1	1
signed char	1	1
float	4	4
double	8	8
long double	10	12

Dev C++ intல் மற்றும் long வகைகளுக்கு 4 பைட்டுகளை ஒதுக்கிறது. Dev C++ல் நிரலை எழுதும் போது பெரிய முழு எண் மதிப்புகளை கையாள்வதற்கு இதில் ஏதேனும் ஒரு வகையை பயன்படுத்தலாம்.

குறிப்பு: sizeof( ) என்னும் செயற்குறி தரவினத்தின் அளவை தருகிறது. (size)

எடுத்துக்காட்டு 9.5: C++ தரவினத்தின் அளவை கண்டறிவதற்கான நிரல்

```
#include <iostream>
using namespace std;
int main()
{
        short a;
        unsigned short b;
        signed short c;
        int d;
        unsigned int e;
        signed int f;
        long g;
        unsigned long h;
        signed long i;
        char j;
        unsigned char k;
        signed char l;
        float m;
        double n;
        long double p;
        cout << "\n Size of short = " << sizeof(a);
        cout << "\n Size of unsigned short = " << sizeof(b);
        cout << "\n Size of signed short = " << sizeof (c);
        cout << "\n Size of int = " << sizeof(d);
        cout << "\n Size of unsigned int = " << sizeof(e);
        cout << "\n Size of signed int = " << sizeof(f);
        cout << "\n Size of long = " << sizeof(g);
        cout << "\n Size of unsigned long = " << sizeof(h);
        cout << "\n Size of signed long = " << sizeof(i);
        cout << "\n Size of char = " << sizeof(j);
        cout << "\n Size of unsigned char = " << sizeof(k);
        cout << "\n Size of signed char = " << sizeof(l);
        cout << "\n Size of float = " << sizeof(m);
        cout << "\n Size of double = " << sizeof(n);
        cout << "\n Size of long double = " << sizeof(p);
}
```

வெளியீடு: **(Dev C++ ல் தொகுத்து இயக்கப்படுகிறது)**

```
        Size of short = 2
        Size of unsigned short = 2
        Size of signed short = 2
        Size of int = 4
        Size of unsigned int = 4
        Size of signed int = 4
        Size of unsigned int = 4
        Size of signed int = 4
```


Chapter 9.indd 36 03/09/18 5:57 PM

```
Size of long = 4
Size of unsigned long = 4
Size of signed long = 4
Size of char = 1
Size of unsigned char = 1
Size of signed char = 1
Size of float = 4
Size of double = 8
Size of long double = 12
```

குறிப்பு:

sizeof( ) எனும் செயற்குறி தரவினத்தின் அளவை வெளியீடாக காட்டும்.

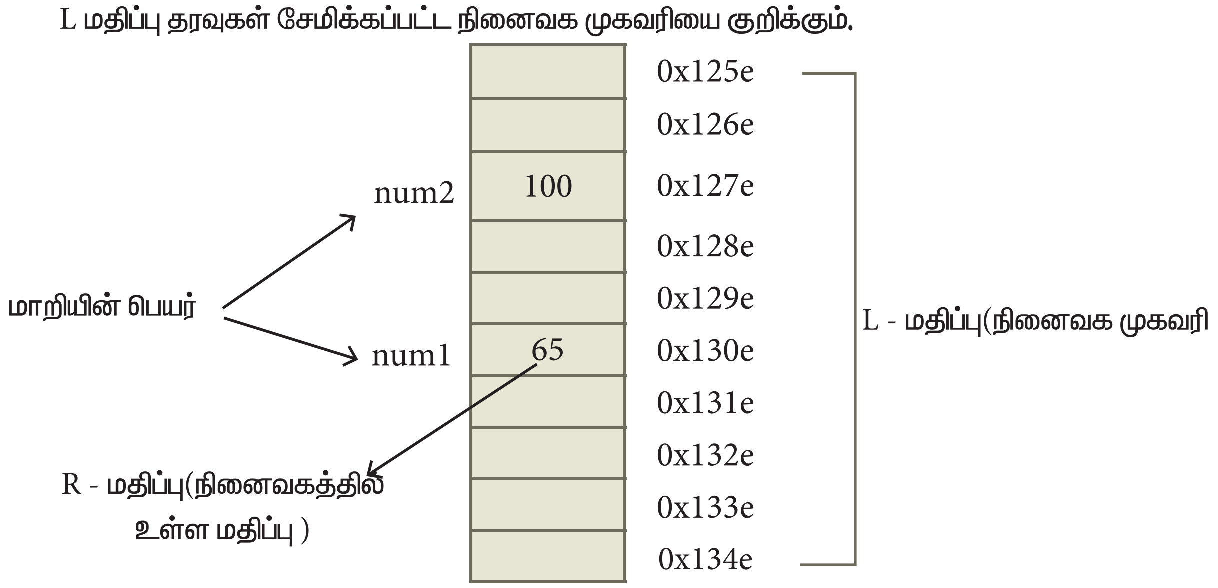
C++ - ல் எண்களின் பின்னொட்டு

முழு எண் மற்றும் மிதப்புப்புள்ளி எண்களுக்கு வேறுபட்ட பின்னொட்டுகள் பயன்படுகின்றன. ஒரே மதிப்பை, பல்வேறு தரவினமாக எடுத்துக்காட்டாக, 45 என்ற மதிப்பை long, unsigned int மற்றும் unsigned long int (ஆங்கில சிறிய அல்லது பெரிய எழுத்து) பின்னொட்டாக சேர்க்க வேண்டும். அதாவது, 45L அல்லது 45U என இவ்வாறு குறிப்பிடுவது, கொடுக்கப்பட்ட மதிப்புகளை long மற்றும் unsigned என எடுத்துக்கொள்ள வேண்டும் என்று நிரல் பெயர்ப்பிக்கு கட்டளையிடுவதாகும். அதேப் போன்று, மிதப்புப் புள்ளி எண்களைக் குறிப்பிட என்ற எழுத்தை பயனபடுத்த வேண்டும். எடுத்துக்காட்டு 9.14

9.13 மாறிகள்

மாறிகள் என்பவை குறிப்பிட்ட தரவு இனங்களின் மதிப்புகளை தாங்கியுள்ள, பயனரால் பெயரிடப்பட்ட நினைவாக இடங்களை குறிக்கிறது. மாறிகள் என்பவை குறிப்பெயர்கள் ஆகும். எனவே, குறிப்பெயர்களுக்கு பெயரிட பயன்படும் விதிமுறைகளையே மாறிகளுக்கு பெயரிடவும் பயன்படுத்த வேண்டும். மாறிகள் பெயரிடப்பட்ட நினைவகங்களை குறிப்பிடுவதால் இவற்றுக்கு குறுயீட்டு மாறிகள் (Symbolic variables) என்று பெயர். R மற்றும் L என்ற இரண்டு மதிப்புகள் குறியீட்டு மாறிகளுடன் தொடர்புடையது.

- R- மதிப்பு நினைவகத்தில் சேமிக்கப்பட்டுள்ள தரவுகளைக் குறிக்கும்.
- L மதிப்பு தரவுகள் சேமிக்கப்பட்ட நினைவக முகவரியை குறிக்கும்.



படம் 9.14 மாறியின் நினைவக ஒதுக்கீடு

Chapter 9.indd 37 03/09/18 5:57 PM

நினைவக முகவரி என்பது பதினாறு நிலை (hexa decimal) எண்களாக உள்ளன என்பதை நினைவில் கொள்க.

9.13.1 மாறிகளின் அறிவிப்பு

அனைத்து மாறிகளும் அதன் நிரல் பயன்பாட்டிற்கு முன்னரே அறிவிக்கப்பட வேண்டும். மாறியின் பெயருடன் கூடிய தரவுவகைக்கு ஏற்ற நினைவக ஒதுக்கீடு பற்றி தொகுப்பானுக்கு தெரிவிப்பதே அறிவிப்பு ஆகும். எடுத்துக்காட்டாக, ஒரு மாறியை int வகையாக DevC++ ல் அறிவித்தால் தொகுப்பான் 4 பைட் நினைவகத்தை ஒதுக்கும். எனவே அனைத்து மாறிகளும் சேமிக்கப்படவேண்டிய மதிப்பின் வகையோடு அறிவிக்கப்பட வேண்டும்.

ஒன்றுக்கு மேற்பட்ட மாறிகளின் அறிவிப்பு

ஒன்றுக்கு மேற்பட்ட ஒரே வகை மாறியை ஒரே கூற்றில் அறிவிக்க முடியும். அக்கூற்றிலுள்ள ஒவ்வொரு மாறியையும் தனித்தனியே குறிக்க காற்புள்ளி (,) பயன்படுத்தப்பட வேண்டும்.

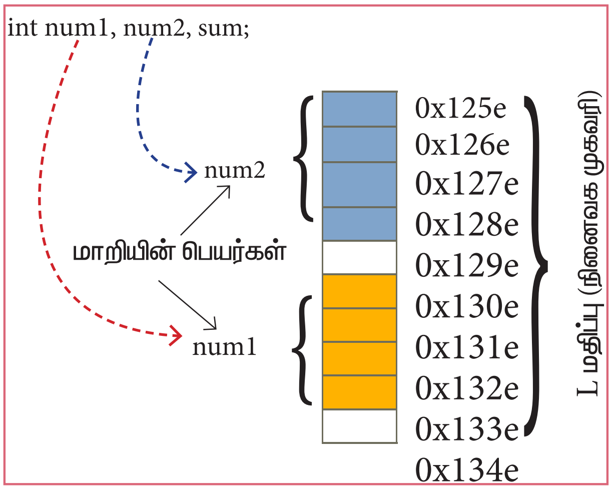
கட்டளை அமைப்பு:

<data type> <var1>, <var2>, <var3> …… <var_n>;

எடுத்துக்காட்டு :

int num1, num2, sum;

மேலே உள்ள கூற்றில் மூன்று முழு எண் வகை மாறிகள் உள்ளன. அதாவது, num1, num2 மற்றும் sum என்ற மாறிகளில் முழு எண் மதிப்புகளை மட்டுமே இருத்த முடியும். இந்த மாறிகளில் மிதப்புப் புள்ளி (தசம மதிப்புகளை சேமிக்க முற்பட்டால் முழு எண் மதிப்பை மட்டுமே நினைவகத்தில் இருத்தப்படும். மதிப்புப் புள்ளி பகுதி ஒதுக்கப்பட்டு விடும். மேலே கூறப்பட்ட அறிவிப்பில் C++ தொகுப்பானானது ஒவ்வொரு மாறிக்கும் 4 பைட்டு நினைவக இடத்தை ஒதுக்கும் (4 நினைவக பெட்டிகள்)



படம்- 9.15 வகை மாறிகளின் நினைவக ஒதுக்கீடு

ஒரு மாறி, தொடக்க மதிப்பு ஏதுமின்றி அறிவிக்கப்பட்டால், அதற்கு ஒதுக்கப்பட்ட நினைவகத்தில் அறியப்படாத மதிப்புகள் இருத்தப்பட்டுவிடும். இவற்றை குப்பை மதிப்புகள் (Junk அல்லது Garbage) என்று பெயர்.

```
#include <iostream>
using namespace std;
int main()
{
        int num1, num2, sum;
        cout << num1 << endl;
        cout << num2 << endl;
        cout << num1 + num2;
}
```

மேலே கொடுக்கப்பட்டுள்ள நிரலில் num1, num2 மாறிகளுக்கு அறியப்படாத மதிப்புகள் நினைவகத்தில் இருத்தப்படும். எனவே cout<< num1+num2; வில் அறியப்படாத குப்பை மதிப்புகளின் கூட்டுத்தொகை விடையாக கிடைக்கும்.

9.13.2 மாறியின் தொடக்க மதிப்பிருத்தல்:

மாறி அல்லது மாறிகள் அறிவிக்கப்படும்போதே அதற்கு தொடக்க மதிப்பு வழங்குதலை "தொடக்க மதிப்பிருத்தல்" என்று அழைக்கப்படுகிறது.

எடுத்துக்காட்டுகள்:

int num = 100;

Chapter 9.indd 38 03/09/18 5:57 PM

float pi = 3.14;

double price = 231.45;

மேலே num, pi மற்றும் price என்ற மாறிகள் அறிவிப்பின் போதே தொடக்க மதிப்பு இருத்தப்பட்டுள்ளன. நிரலின் இயக்கத்தின் போது இந்த மாறிகளின் தொடக்க மதிப்பை மாற்றிக் கொள்ளலாம்.

எடுத்துக்காட்டு **9.6** உருளையின் வளைந்த மேற்பரப்பை கண்டறியும் நிரல் **(CSA = 2 pi rh)**

```
#include <iostream>
using namespace std;
int main()
{
        float pi = 3.14, radius, height, CSA;
        cout << "\n Curved Surface Area of a cylinder";
        cout << "\n Enter Radius (in cm): ";
        cin >> radius;
        cout << "\n Enter Height (in cm): ";
        cin >> height;
        CSA = (2*pi*radius)*height;
        system("cls");
        cout << "\n Radius: " << radius <<"cm";
        cout << "\n Height: " << height << "cm";
        cout << "\n Curved Surface Area of a Cylinder is " << CSA <<" sq. cm.";
}
```

வெளியீடு:

```
        Curved Surface Area of a cylinder
        Enter Radius (in cm): 7
        Enter Height (in cm): 20
        Radius: 7cm
        Height: 20cm
        Curved Surface Area of a Cylinder is 879.2 sq. cm.
```

ஒரே வகையுள்ள மாறிகளுக்கு ஒரே கூற்றில் தொடக்க மதிப்பிருத்த முடியும்.

எடுத்துக்காட்டு:

int x1 = -1, x2 = 1, x3, n;

9.13.3 இயங்கு நிலை தொடக்க மதிப்பிருத்தல்

நிரலின் இயக்கத்தின் போது ஒரு மாறிக்கு தொடக்க மதிப்பு இருத்த முடியும். இதுவே இயங்குநிலை தொடக்க மதிப்பிருத்தல் எனப்படும்.

எடுத்துக்காட்டு

int num1, num2, sum;

Chapter 9.indd 39 03/09/18 5:57 PM

sum = num1 + num2;

மேற்கண்ட இரண்டு கூற்றுகளும் ஒரே கூற்றாக பின்வருமாறு இணைக்கப்பட்டுள்ளது.:

int sum = num1+num2;

இயக்கத்தின் போது அறியப்பட்ட num1 மற்றும் num 2. மதிப்புகளை பயன்படுத்தி sum க்கு தொடக்க மதிப்பு கருத்தப்படுகிறது.

எடுத்துக்காட்டு 9.7இயங்கு நிலை தொடக்க மதிப்பிருத்தலை விளக்கும் C++ நிரல்

```
#include <iostream>
using namespace std;
int main()
{
        int num1, num2;
        cout << "\n Enter number 1: ";
        cin >> num1;
        cout << "\n Enter number 2: ";
        cin >> num2;
        int sum = num1 + num2; // Dynamic initialization
        cout << "\n Average: " << sum /2;
}
```

வெளியீடு:

```
        Enter number 1: 78
        Enter number 2: 65
        Average: 71
```

மேலே கொடுக்கப்பட்ட நிரலில் num1 மற்றும் num2 மாறிகளுக்க மதிப்புகளை பெற்ற பின் sum என்ற மாறி அறிவிக்கப்பட்டு இந்த இரண்டு மாறிகளின் கூட்டுத் தொகை மதிப்பிருத்தப்படுகிறது பிறகு அது இரண்டால் வகுக்கப்பட்டு வெளியீடு காட்டப்படுகிறது.

எடுத்துக்காட்டு 9.8 அரைவட்டத்தின் சுற்றளவு மற்றும் பரப்பை கண்டறியும் C++ நிரல்

```
#include <iostream>
using namespace std;
int main()
{
        int radius;
        float pi = 3.14;
        cout << "\n Enter Radius (in cm): ";
        cin >> radius;
        float perimeter = (pi+2)*radius;        // dynamic initialization
        float area = (pi*radius*radius)/2; // dynamic initialization
        cout << "\n Perimeter of the semicircle is " << perimeter << " cm";
        cout << "\n Area of the semicircle is " << area << " sq.cm";
}
```

வெளியீடு:

Enter Radius (in cm): 14
Perimeter of the semicircle is 71.96 cm
Area of the semicircle is 307.72 sq.cm

9.13.4 அணுகுநிலை பண்புணர்த்தி const;

மாறிலியை அறிவிப்பதற்கான சிறப்பு சொல் const ஆகும். const சிறப்பு சொல் மாறியின் அணுகுநிலையை மாற்றுகிறது அல்லது முறைப்படுத்துகிறது. எனவே இது அணுகுநிலை பண்புணர்த்தி எனவும் அழைக்கப்படுகிறது.

எடுத்துக்காட்டு:

int num = 100;

மேலே உள்ள கூற்று num1 என்ற மாறியை 100 என்ற தொடக்க மதிப்போடு அறிவிக்கிறது. எனினும் இயக்கத்தின் போது num ன் மதிப்பு மாற்றப்படலாம். மேலே உள்ள கூற்றை, const int num=100; என மாற்றும் போது, மாறி num என்பது மாறிலியாகிறது. அதனுடைய மதிப்பு இந்த நிரல் முழுவதும் 100 என இருக்கும், இயக்கத்தின் போது மாறாது.

```
#include <iostream>
using namespace std;
int main()
{
        const int num=100;
        cout << "\n Value of num is = " << num;
        num = num + 1; // Trying to increment the constant
        cout << "\n Value of num after increment " << num;
}
```

மேலே கொடுக்கப்பட்டுள்ள நிரல் Dev C++ ல் இயங்கும் போது “assignment of read only memory num” என்ற பிழை செய்தியும், Turbo C++ ல் நிரல் பெயர்ப்பியில் இயங்கும் போது, “Cannot modify the const object” மற்றும் என்ற பிழை செய்தியும் காட்டப்படும்.

Chapter 9.indd 41 03/09/18 5:57 PM

சுய மதிப்பீடு

1. பண்புணர்த்தி என்றால் என்ன? அதன் பயன் யாது?
2. பின்வரும் C++ கூற்றிலுள்ள பிழை என்ன?

   long float x;
3. மாறி என்றால் என்ன? மாறியை ஏன் குறியீட்டு மாறி என்று அழைக்கிறோம்?
4. இயங்குநிலையில் மாறியை தொடங்குதல் என்றால் என்ன? எடுத்துக்காட்டு தருக
5. கீழ்கண்ட கூற்றில் உள்ள பிழை யாது?

   const int x;

9.13.5 குறிப்புகள் (References):

முன்னரே, வரையறுக்கப்பட்ட மாறிகளுக்கு ஒரு மறுபெயரை குறிப்புகள் வழங்குகின்றன. குறிப்புகளின் அறிவிப்பு மாறியின் அடிப்படை தரவினத்துடன் & குறியீட்டையும் கொண்டிருக்கும். குறிப்பு மாறியின் பெயரானது ஏற்கனவே அறிவிக்கப்பட்ட மாறியின் மதிப்பை எடுத்துக் கொள்ளும்.

கட்டளை அமைப்பு:

<type> <& reference_variable> = <original_variable>

எடுத்துக்காட்டு 9.9 குறிப்பு மாறிகளை அறிவிக்கும் C++ நிரல்

```
#include <iostream>
using namespace std;
int main()
{
        int num;
        int &temp = num; //declaration of a reference variable temp
        num = 100;
cout << "\n The value of num = " << num;
cout << "\n The value of temp = " << temp;
}
```

வெளியீடு:

```
        The value of num = 100
        The value of temp = 100
```

9.14 வெளியீடு வடிவமைப்பு:

எளிதாக படிப்பதற்கும் மற்றும் புரிந்து கொள்வதற்கும் தேவைப்படும் வெளியீட்டு திரையினை உருவாக்குவதற்கு வெளியீடு வடிவமைப்பு மிக முக்கியமானதாகும். C++ நிரல் வெளியீடுகளை வடிவமைக்க கையாளுகைகள் என்பது (Manipulation) பயன்படுகிறது. கையாளுகைகள் என்பது, தரவு பெறும் (<<)

Chapter 9.indd 42 03/09/18 5:57 PM

மற்றும் தரவு ஈர்ப்பு (>>) செயற்குறிகளில் பயன்படுத்த சிறப்பாக வடிவமைக்கப்பட்ட செயற்கூறுகள் ஆகும்.

வெளியீட்டை வடிவமைக்க C++ பல்வேறு கையாளுகைகளை வழங்கியுள்ளது. endl set w, set fill, set precision மற்றும் set f போன்றவை பொதுவாக பயன்படுத்தப்படும். கையாளுகை பெயற்கூறுகளாகும். இச் செயற்கூறுகளை பயன்படுத்த பொருத்தமான தலைப்புக் கோப்புகளை நிரலுடன் இணைக்க வேண்டும். endl கையாளுகை iostream தலைப்புக் கோப்பின் உறுப்பு ஆகும். set w, set fill, set precision மற்றும் set f கையாளுகைகள் iomanip என்ற தலைப்புக் கோப்பின் உறுப்பு ஆகும்.

endl (வரியின் முடிவு)

C++ -ல் endl என்பது வரி உள்ளீடாக பயன்படுகிறது. இது '\n'க்கு மாற்றாக பயன்படுகிறது. endl ஒரு புதிய வரியை செருகி, செருகும் புள்ளியை அந்த வரியின் தொடக்கத்தில் நிறுத்துகிறது. endl மற்றும் '\n', ஒரே மாதிரியான வேலைகளை செய்தாலும் இதற்கிடைய ஒரு வேறுபாடு உள்ளது.

- endl – புதிய வரியை செருகி தற்காலிக நினைவகத்தை காலி செய்கிறது
- '\n' - புதிய வரியை மட்டும் செருகுகிறது.

எடுத்துக்காட்டு:

cout << "**\n** The value of num = " << num;

cout << "The value of num = " << num <<**end;**

இந்த இரண்டு கூற்றுகளும் ஒரே வெளியீட்டை தருகின்றன.

setw ( )

setw() கையாளுகை செயற்கூறு வெளியீட்டிற்காக ஒதுக்கப்பட்ட புலத்தின் அகலத்தை வரையறுக்கிறது. வெளியீட்டில் எழுதப்பட வேண்டிய குறைந்தபட்ச குறியுரு எண்ணிக்கையை புலத்தின் அகலம் நிர்ணயிக்கிறது.

கட்டளையமைப்பு:

set w (எழுத்துருக்களின் எண்ணிக்கை)

எடுத்துக்காட்டு **9.10** சம்பளக் கணக்கீட்டிற்கான நிரல்

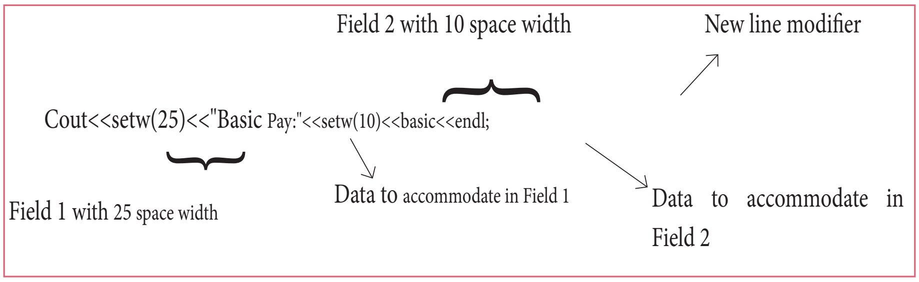
```
#include <iostream>
#include <iomanip>
using namespace std;
int main()
{
        float basic, da, hra, gpf, tax, gross, np;
        char name[30];
        cout << "\n Enter Basic Pay: ";
        cin >> basic;
        cout << "\n Enter D.A : ";
        cin >> da;
        cout << "\n Enter H.R.A: ";
        cin >> hra;
```

```
        gross = basic+da+hra; // sum of basic, da nad hra
        gpf = (basic+da) * 0.10; // 10% 0f basic and da
        tax = gross * 0.10; //10% of gross pay
        np = gross - (gpf+tax); //netpay = earnings - deductions
        cout << setw(25) << "Basic Pay : " << setw(10)<< basic<< endl;
        cout << setw(25) << "Dearness Allowance : "<< setw(10)<<da<< endl;
        cout<<setw(25)<<"House Rent Allowance : "<<setw(10)<< hra<<endl;
        cout << setw(25) << "Gross Pay : " << setw(10) << gross << endl;
        cout << setw(25) << "G.P.F : " << setw(10) << gpf << endl;
        cout << setw(25) << "Income Tax : " << setw(10)<< tax << endl;
        cout << setw(25) << "Net Pay : " << setw(10) << np << endl;
}
```

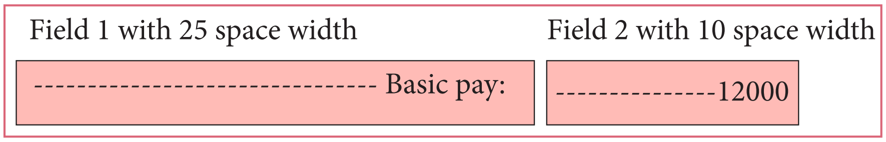
வெளியீடு:

```
Enter Basic Pay: 12000
Enter D.A : 1250
Enter H.R.A: 1450
               Basic Pay  :     12000
      Dearness Allowance  :      1250
    House Rent Allowance  :      1450
               Gross Pay  :     14700
                   G.P.F  :      1325
              Income Tax  :      1470
                 Net Pay  :     11905
```

மேலேகொடுக்கப்பட்டநிரலில்,ஒவ்வொருவெளியீட்டுக்கூற்றும்இரண்டுsetw( )கையாளுகைகளைக் கொண்டுள்ளது. முதலில் கொடுக்கப்பட்டுள்ள setw(25) என்பது 25 இடைவெளிகளைக் கொண்ட ஒரு புலத்தையும், setw(10) என்பது 10 இடைவெளிகளையும் கொண்ட மற்றொரு புலத்தையும் உருவாக்குகின்றது. இப்புலங்களில் மதிப்புகளை இருத்தும் போது, அவை வலமிருந்து இடமாக தோன்றும்.



புலம் 1 மற்றும் புலம் 2 –ல் சரம் “Basic Pay: ” மற்றும் value of basic pay வின் மதிப்பு



படம் 9.16 setw( ) செயற்கூறு

Chapter 9.indd 44
03/09/18 5:57 PM

setfill ( )

இந்த கையாளுகை செயற்கூறு பொதுவாக setw விற்கு பிறகு பயன்படுகிறது. கொடுக்கப்பட்ட மதிப்பு பொறுத்தப்பட்ட அகலத்தை முழுவதுமாக நிரப்பவில்லையெனில், setfill அளபுருவில் குறிப்பிடப்பட்டுள்ள குறியுரு புலத்தில் காலியாக உள்ள இடங்களை நிரப்பப் பயன்படுகிறது.

கட்டளையமைப்பு:

setfill (எழுத்துருக்கள்);

எடுத்துக்காட்டு:

cout << "\n H.R.A : " << setw(10) << setfill (0) << hra;

மேலே கொடுக்கப்பட்ட குறிமுறையில் setw கொடுக்கப்பட்ட மதிப்பை காண்பிக்க ஒரு புலத்தை உருவாக்குகிறது. Setfill நிரப்பபட்டாமல் காலியாக உள்ள இடத்தை 0 (zero)ல் நிரப்புகிறது.

எடுத்துக்காட்டாக, hra வின் மதிப்பு 1200 என கொடுத்தால் setw என்பது 10 இடைவெளி புலத்தில் வலதிலிருந்து இடது புறமாக 1200 ஐ இருத்துகிறது. setfill என்பது முதலில் வருகின்ற காலியாகமீதமுள்ள இடங்களில் 0வைநிரப்புகிறது. வெளியீடானது, 0000001200 என்று வரும்.

setprecision ( )

கொடுக்கப்பட்ட இலக்கங்களின் எண்ணிக்கைக்கு இணையான எண்களை தசம எண்களாக காண்பிக்க பயன்படுகிறது..

கட்டளை அமைப்பு:

setprecision (இலக்கங்களின் எண்ணிக்கை);

எடுத்துக்காட்டு:

```
#include <iostream>

#include <iomanip>

using namespace std;

int main()

{

    float hra = 1200.123;

    cout << setprecision (5) << hra;

}
```

மேலே கொடுக்கப்பட்ட குறிமுறையில் கொடுக்கப்பட்ட மதிப்பு 1200.123 தசமத்தையும் சேர்த்து 5 இலக்கமாக காண்பிக்கப்படும். எனவே, வெளியீடு 1200.1 எனத் தோன்றும்.

setprecision ( ) இடமிருந்து வலது புறமாக மதிப்புகளை அச்சிடும். மேலே கொடுக்கப்பட்ட குறிமுறையில் முதலில் 4 இலக்கங்கள் எடுத்துக் கொள்ளப்பட்டு பின்பு தசம பகுதியில் ஒரு இலக்கம் அச்சிடப்படும்.

setprecision காண்பிக்கப்பட வேண்டிய தசம இடங்களின் எண்ணிக்கையை பொறுத்துவதற்கு உதவுகிறது. இந்த வேலையை செய்வதற்காக setf() கையாளுகை செயற்கூறில் ios flag ஐ பொறுத்த வேண்டும். இது இரண்டு வடிவங்களில் பயன்படுத்தப்படும். அவை, (i) நிலையான (fixed) மற்றும் (ii) அறிவியல் (scientific)

setprecision கையாளுகைக்கு முன்னர், fixed அல்லது Scientific என்ற இரண்டு சிறப்புச் சொற்களில் பொருத்தமானதை பயன்படுத்த வேண்டும்.

எடுத்துக்காட்டு :

```
#include <iostream>

#include <iomanip>

using namespace std;

int main()
```


Chapter 9.indd 45 03/09/18 5:57 PM

```
{
        cout.setf(ios::fixed);
        cout << setprecision(2)<<0.1;
}
```

மேலே கொடுக்கப்பட்டுள்ள நிரலில் , ios flagல் fixed வகை பொறுத்தப்பட்டுள்ளது. இது தசமப் புள்ளி எண்களை நிலையான வடிவில் அச்சிடுகிறது. எனவே, விடையானது, 0.10என்று கிடைக்கும்.

```
cout.setf(ios::scientific);
cout << setprecision(2) << 0.1;
```

மேலே கொடுக்கப்பட்டுள்ள நிரலில் , ios flag - scientific வகையாக பொறுத்தப்படுகிறது. இது தசமப் புள்ளி எண்களை scientific வடிவில் அச்சிடுகிறது. எனவே, இதன் வெளியீடு 1.00e-001

9.15 கோவை (Expression):

கோவை என்பது C++ விதிமுறைக்குட்பட்ட சீராக உள்ள செய்முறைகள், மாறிலிகள் மற்றும் மாறிகளின் கூட்டாகும். இது திருப்பி அனுப்பும் மதிப்பையுடைய செயற்கூறுகளையும் உள்ளடக்கியதாகும். செயற்கூறுகளைப் பற்றி பின்வரும் பாடங்களில் காணலாம்.

கோவை ஒரு மதிப்பை உருவாக்க ஒன்றுக்கு மேற்பட்ட செயலேற்பிகள் மற்றும் சுழியம் அல்லது அதற்கு மேற்பட்ட செயற்குறிகள் ஆகியவற்றை உள்ளடக்கியது. C++-ல் , ஏழு விதமான கோவைகள் உள்ளன. அவையாவன:

(i) மாறிலிக் கோவை

(ii) முழு எண் கோவை

(iii) மிதப்புப் புள்ளி கோவை

(iv) ஒப்பீட்டுக் கோவை

(v) தருக்கக் கோவை

(vi) பிட்நிலை கோவை

(vii) சுட்டுக் கோவை

வ.எண்	கோவை	விளக்கம்	எடுத்துக்காட்டு
1	மாறிலிக் கோவை	மாறிலிக் கோவை மாறிலி மதிப்புகளை மட்டுமே உள்ளடக்கியது	int num=100;
2	முழு எண் கோவை	முழு எண் மற்றும் குறியுரு மதிப்புகள் அல்லது மாறிலிகள் எளிய கணக்கீட்டு செயற்குறிகளுடன் இணைந்து முழு எண் விடையை தருகின்றன.	sum=num1+num2; avg=sum/5;
3	மிதப்புப் புள்ளி கோவை	மிதப்புப் புள்ளி மதிப்புகள் அல்லது மாறிகள் எளிய கணக்கீட்டுச் செயற்குறிகளுடன் இணைந்து மிதப்புப் புள்ளி விடையை தருகின்றன.	Area=3.14*r*r;
4	ஒப்பீட்டுக் கோவை	மதிப்புகள் அல்லது மாறிலிகள் ஒப்பீட்டு செயற்குறிகளுடன் இணைந்து பூலியன் (சரி அல்லது தவறு) மதிப்புகளை விடையாக கொடுக்கும்.	x>y; a+b==c+d;

Chapter 9.indd 46 03/09/18 5:57 PM

5	தருக்கக் கோவை	மதிப்புகள் அல்லது மாறிலிகள் தருக்கச் செயற்குறிகளுடன் இணைந்து பூலியன் மதிப்புகளை விடையாகக் கொடுக்கும்.	(a>b)&& (c==10);
6	பிட் நிலை கோவை	மதிப்புகள் அல்லது மாறிகள் பிட் நிலை செயற்குறிகளுடன் இணைந்து இந்த கோவை உருவாகிறது.	x>>3; a<<2;
7	சுட்டுக் கோவை	சுட்டு என்பது நினைவக முகவரியை ஏற்கும் மாறி ஆகும்.	int *ptr;

அட்டவணை 9.10 : கோவைகளின் வகைகள்

9.16 இன மாற்றம் (Type Conversion)

ஒரு அடிப்படை இனத்திலிருந்து மற்றொரு இனத்திற்கு மாற்றப்படும் முறையே இனமாற்றம் எனப்படும். C++ இரண்டு வகையான இனமாற்றத்தை வழங்குகிறது.

(1) உள்ளுறை இனமாற்றம் மற்றும் (2) வெளியுறை இனமாற்றம்

(1) உள்ளுறை இனமாற்றம் (Implicit type conversion):

உள்ளுறை இனமாற்றம் என்பது தொகுப்பான் தானாகவே மாற்றத்தை ஏற்படுத்திக் கொள்வதாகும். இது தானியங்கி மாற்றம் (“Automatic conversion)” என்றும் அழைக்கப்படும்.

இந்த வகையாக இனமாற்றம் ஒரு கோவையில் வேறுபட்ட தரவினங்கள் கலந்திருக்கும் போது உபயோகிக்கப்படுகின்றன. செயலேற்பிகளின் வகை மாறுபடும் போது தொகுப்பானானது அவற்றில் ஒன்றை மற்றொன்றிக்கு இணையாகுமாறு மாற்றுகிறது. விதிமுறைகளின் படி சிறிய வகை பெரிய வகைக்கு மாற்றப்படுகிறது. இதுவே, இன உயர்வு “Type Promotion” எனப்படும்.

எடுத்துக்காட்டு:

```
#include <iostream>
using namespace std;
int main()
{
    int a=6;
    float b=3.14;
    cout << a+b;
}
```

மேலே கொடுக்கப்பட்ட நிரலில் செயலேற்பி **a** என்பது, ஒரு முழு எண் வகை, **b** என்பது ஒரு மிதப்புப் புள்ளி வகை. நிரலின் இயக்கத்தின் போது முழு எண் மிதப்புப் புள்ளியாக மாற்றப்படுகிறது. ஏனெனில், மிதப்புப் புள்ளி, முழு எண்ணை விட பெரிய வகையாகும். எனவே, மேலே கொடுக்கப்பட்ட நிரலின் விடையானது 9.14 – என்று கிடைக்கும்.

அட்டவணை 9.11 இனமாற்ற அமைப்பை காட்டுகிறது.

அட்டவணை 9.11: கலப்பு செயலேற்பிகளின் உள்ளுறை இனமாற்றம்

Chapter 9.indd 47 03/09/18 5:57 PM

RHO / LHO	**char**	**short**	**int**	**long**	**float**	**double**	**long double**
char	int	int	int	long	float	double	long double
short	int	int	int	long	float	double	long double
int	int	int	int	long	float	double	long double
long	long	long	long	long	float	double	long double
float	float	float	float	float	float	double	long double
double	double	double	double	double	double	double	long double
long double	long double	long double	long double	long double	long double	long double	long double

(RHO – Right Hand Operand; LHO – Left Hand Operand)

(2) வெளியுறை இனமாற்றம் (Explicit type conversion)

மாறிகள் அல்லது கோவைகளை ஒரு தரவினத்திலிருந்து மற்றொரு குறிப்பிட்ட தரவினத்திற்கு பயனரை மாற்றம் செய்ய அனுமதிப்பதே வெளியுறை இனமாற்றம் எனப்படும். இதுவே, இனவார்ப்புரு "type casting" என்று அழைக்கப்படுகிறது.

கட்டளையமைப்பு:

(type-name) expression;

இங்கே, இனப்பெயர் என்பது மாற்றம் ஏற்படுத்துவதற்கான தகுதி வாய்ந்த C++ தரவினம் ஆகும்.

எடுத்துக்காட்டு

```
#include <iostream>

using namespace std;
int main( )
{
    float varf=78.685;
    cout << (int) varf;
}
```

மேலே கொடுக்கப்பட்டுள்ள நிரலில், varf என்ற மாறியானது 78.685 என்ற முதல் மதிப்போடு மிதப்புப் புள்ளி (float) தரவினமாக அறிவிக்கப்பட்டுள்ளது. வெளியீடு கூற்றில் varf ன் மதிப்பு முழு எண் தரவினமாக வெளியுறை இனமாற்றம் செய்யப்படுகிறது. இறுதி வெளியீடு 78 ஆகும்.

வெளியுறை இனமாற்றத்தின்போது, அதிக பரப்பு கொண்ட வகைக்கு மதிப்பை கொடுக்கும் போது அது எந்த சிக்கலையும் ஏற்படுத்தாது. ஆனால் பெரிய வகை மதிப்பை சிறிய வகையில் கொடுக்கும் போது சில துல்லியமான மதிப்புகளில் இழப்பு ஏற்படலாம்.

வ.எண்	வெளியுறை இனமாற்றம்	சிக்கல்
1	இரட்டை மிதப்புப் புள்ளியிலிருந்து மிதப்புப்புள்ளி(double to float)	துல்லியத்தில் இழப்பு. அசல் மதிப்பு இலக்க வகையின் வரையெல்லையை மீறினால் விடையை வரையறுக்க முடியாது.
2	மிதப்புப் புள்ளியிலிருந்து முழு எண்(float to int)	தசமப் பகுதியின் இழப்பு அசல் மதிப்பு இலக்க வகையின் வலையெல்லையை மீறினால் விடையை வரையறுக்க முடியாது.
3	long to short	தரவு இழப்பு

அட்டவணை 9.12 – வெளியுறை இனமாற்ற சிக்கல்கள்

எடுத்துக்காட்டு:

```
#include <iostream>
using namespace std;
int main()
{
        double varf=178.25255685;
        cout << (float) varf << endl;
        cout << (int) varf << endl;
}
```

வெளியீடு:

```
178.253
178
```

சுய மதிப்பீடு

1. இனமாற்றம் என்றால் என்ன?
2. உள்நிலை மாற்றம் வெளிநிலை மாற்றத்திலிருந்து எவ்வாறு வேறுபடுகிறது?
3. endl மற்றும் \n வேறுபடுத்துக
4. குறிப்புரையின் பயன் யாது?
5. Setprecision() –பயன் யாது?

Chapter 9.indd 49 03/09/18 5:57 PM

செய்முறைப் பயிற்சி:

1. இரண்டு மாறிகளின் மதிப்பை ஒன்றிலிருந்து மற்றொன்றுக்கு மாற்றுவதற்கான C++ நிரலை எழுதுக.
   a. மூன்றாவது மாறியைப் பயன்படுத்தி
   b. மூன்றாவது மாறியை பயன்படுத்தாமல்
2. பின்வருவனவற்றை செய்வதற்கு C++ நிரலை எழுதுக:
   a. வட்டத்தின் கால்பகுதிக்கான சுற்றளவு மற்றும் பரப்பளவை கண்டறிக.
   b. முக்கோணத்தின் பரப்பளவை கண்டறிக.
   c. வெப்பநிலையை செல்ஸியஸ்லிருந்து, ஃபாரன்ஹீட்டாக மாற்றுக.
4. உன்னுடைய பத்தாம் வகுப்பு பொதுத்தேர்வு மதிப்பெண்களின் கூட்டுத்தொகை மற்றும் சராசரியை கண்டறிந்து, அனைத்து மதிப்புகளையும் ஒன்றன் பின் ஒன்றாக கூட்டுத்தொகை மற்றும் சராசரியுடன் காண்பிக்கவும். வடிவமைப்பு செயற்கூறுகளை பயன்படுத்தவும்.

நினைவில் கொள்க

- அனைத்து நிரலாக்க மொழிகளிலும் இரண்டு அடிப்படை உறுப்புகள் உள்ளன.
- C++-ல் தரவு இனங்களை மூன்று வகைகளாக பிரிக்கலாம். (1)அடிப்படைத் தரவு இனங்கள் (2) பயனர் வரையறுக்கும் தரவு இனங்கள் மற்றும் (3) தருவிக்கப்பட்ட தரவு இனங்கள்
- மாறிகள் என்பவை குறிப்பிட்ட தரவு இனங்களின் மதிப்புகளை தாங்கியுள்ள, பெயரிடப்பட்ட நினைவக இடங்களை குறிக்கிறது.
- C++-ல் ஐந்து அடிப்படை தரவினங்கள் உள்ளன. அவை char, int, float, double மற்றும் void.
- தொகுப்பானின் தரவரிசைப்படி ஒவ்வொரு தரவிற்கும் குறிப்பிட்ட நினைவக இடத்தை C++ தொகுப்பான் ஒதுக்குகிறது .
- மாறியின் பெயருடன் கூடிய தரவு வகைக்கு ஏற்ற நினைவக ஒதுக்கீடு பற்றி தொகுப்பானுக்கு தெரிவிப்பதே அறிவிப்பு ஆகும்.
- நிரலின் வெளியீட்டை வடிவமைப்பதற்கு கையாளுகை செயற்கூறுகள் பயன்படுகின்றன. கையாளுகை செயற்கூறுகள் என்பது தரவு விடுப்பு(<<) மற்றும்தரவுஈர்ப்பு(>>) செயற்குறிகளுடன் பயன்படுத்துவதற்கு சிறப்பாக வடிவமைக்கப்பட்ட செயற்கூறுகள் ஆகும்.
- கோவை என்பது C++ விதிமுறைகளுக்குட்பட்ட சீராக உள்ள செய்முறைகள், மாறிலிகள் மற்றும் மாறிகளின் கூட்டாகும்.
- ஒரு அடிப்படை இனத்திலிருந்து மற்றொரு இனத்திற்கு மாற்றப்படும் முறையே இனமாற்றம் எனப்படும். C++ இரண்டு வகையான இனமாற்றத்தை வழங்குகிறது.
- உள்ளுறை இனமாற்றம்
- வெளியுறை இனமாற்றம்

Chapter 9.indd 50 03/09/18 5:57 PM

மதிப்பீடு

MZYH96

பகுதி-அ

சரியான விடையை தேர்வு செய்யவும் :

1. C++-ல் எத்தனை வகையான தரவினங்கள் உள்ளன?

   (அ) 5 (ஆ) 4 (இ) 3 (ஈ) 2

2. பின்வருவனவற்றுள் எது அடிப்படை தரவினம் அல்ல?

   (அ) signed (ஆ) int (இ) float (ஈ) char

3. பின்வரும் கூற்றுகளின் விடையை கண்டறிக?

```
char ch= 'B';
cout << (int) ch;
```

   (அ) B (ஆ) b (இ) 65 (ஈ) 66

4. மிதப்புப் புள்ளி மதிப்பை குறிப்பதற்கு பின்னொட்டாக பயன்படும் குறியுரு எது?

   (அ) F (ஆ) C (இ) L (ஈ) D

5. Dev C++-ல், short int x; என்ற கூற்றில் மாறியில் அறிவிப்புக்கு எத்தனை பைட்டுகள் நினைவகத்தில் ஒதுக்கப்படும்

   (அ) 2 (ஆ) 4 (இ) 6 (ஈ) 8

6. பின்வரும் கூற்றுகளின் வெளியீட்டை கண்டறிக.

```
char ch = 'A';
ch = ch + 1;
```

   (அ) B (ஆ) A1 (இ) F (ஈ) 1A

7. பின்வருவனவற்றுள் எது தரவினங்களின் பண்புணர்த்தி அல்ல?

   (அ) signed (ஆ) int (இ) long (ஈ) short

8. பின்வரும் செயற்குறிகள் எது தரவினங்களின் அளவை தருகிறது?

   (அ) sizeof( ) (ஆ) int ( ) (இ) long ( ) (ஈ) double ( )

9. எந்த செயற்குறி மாறியின் முகவரியை பெற பயன்படுகிறது?

   (அ) $ (ஆ) # (இ) & (ஈ) !

10. endl கட்டளைக்கு மாற்றாக பயன்படுவது எது?

(அ) \t (ஆ) \b (இ) \0 (ஈ) \n

பகுதி-ஆ

குறு வினாக்கள் (2 மதிப்பெண்கள்) :

1. const சிறப்பு சொல் பற்றி எடுத்தக்காட்டுடன் சிறுகுறிப்பு எழுதுக.

2. setw( ) வடிவமைப்பு கையாளும் செயற்கூறின் பயன் என்ன?

3. குறியுரு(char)தரவினம் ஏன் முழு எண் தரவினமாக கருதப்படுகிறது?

4. மேற்கோள் மாறிகள் என்றால் என்ன?அதன் பயன் யாது?

5. பின்வரும் C++ கூற்றுகள் சமமானதா என்பதை கண்டறிக.

   char ch = 67;

   char ch = 'C';

6. 56L மற்றும் 56 இடையே உள்ள வேறுபாடு என்ன?

7. பின்வருவனவற்றுள் எது தகுதி வாய்ந்த மாறிலி என கண்டறிந்து, அதனுடைய வகையை குறிப்பிடுக.

   (i) 0.5 (ii) 'Name' (iii) '\t' (iv) 27,822

8. x மற்றும் y என்பது இரண்டு இரட்டை மிதப்புப் புள்ளி மாறி என்றால் அதனை முழு எண்ணாக மாற்ற பயன்படும் C++ கூற்றை எழுதுக.

9. num=6 என்று முதலில் கொடுக்கப்பட்டால் பின்வரும் கூற்றின் விடையை காண்க.

   (a) cout << num;

   (b) cout << (num==5);

10. பின்வரும் இரண்டு கூற்றில் எது தகுதியானது என கண்டறிந்து அதன் விடையை எழுதுக.

   int a;

   (i) a = 3,014; (ii) a=(3,014);

பகுதி-இ

சிறு வினாக்கள் (3 மதிப்பெண்கள்) :

1. C++-ல் கணக்கீட்டுச் செயற்குறிகள் யாவை? ஒரும, இரும செயற்குறிகளை எடுத்துக்காட்டுடன் வேறுபடுத்துக.

2. மதிப்பிடுக x+= x + ++x; Let x=5;

3. ஒப்பீட்டு செயற்குறிகளும், தருக்கச் செயற்குறிகளும் எந்த வகையில் தொடர்புடையவை?

Chapter 9.indd 52 03/09/18 5:57 PM

4. பின்வரும் C++ கோவையை மதிப்பிடுக. இங்கு x, y, z என்பது முழு எண்கள் மற்றும் m, n என்பது மிதப்புப் புள்ளி எண்கள் x = 5, y = 4 மற்றும் m=2.5;

(i) n = x + y / x;

(ii) z = m * x + y;

(iii) z = (x++) * m + x;

துணை நூல்கள்:

(1) Object Oriented Programming with C++ (4th Edition), Dr. E. Balagurusamy, Mc.Graw Hills.

(2) The Complete Reference C++ (Forth Edition), Herbert Schildt. Mc.Graw Hills.

(3) Computer Science with C++ (A text book of CBSE XI and XII), Sumita Arora, Dhanpat Rai & Co.

பாய்வுக் கட்டுப்பாடு

கற்றலின் நோக்கங்கள்

இந்தப் பாடப்பகுதியைக் கற்றபின் மாணவர் அறிந்து கொள்வது.

- கட்டுப்பாடு கூற்றை பற்றி அறிந்து கொள்ளுதல்.
- தேர்ந்தெடுப்பு கூற்றுக்கும், மடக்கு கூற்றுக்கும் உள்ள வேறுபாட்டினை தெரிந்து கொள்ளுதல்.
- if மற்றும் switch கூற்றினை புரிந்து கொள்ளுதல்.
- for, while, do while மடக்கு போன்ற பல வகையான மடக்கு கூற்றுகளை புரிந்துகொள்ளுதல்.

10.1 அறிமுகம்

C++ நிரலாக்கத்தின் மாறிகள், மாறிலிகள், செயற்குறிகள், தரவினங்கள் போன்றவற்றின் அடிப்படை கருத்துகளைப் பற்றி முந்தைய பாடப்பகுதியில் பார்த்தோம். பொதுவாக ஒரு நிரலின் கூற்றுகள் தொடக்கத்திலிருந்து இறுதி வரை ஒன்றன் பின் ஒன்றாக நிறைவேற்றப்படும். இருப்பினும் இத்தகைய வரிசைமுறை இயக்கம் கட்டுப்பாடுடையதாகவும் குறைந்த பயனுள்ளதாகவும் உள்ளது. பல சூழ்நிலைகளில் குறிப்பிட்ட நிபந்தனையின் 1 பி+அடிப்படையில் நிரல் குறிமுறையின் தொகுதி நிறைவேற்றப்பட வேண்டும் என்று தீர்மானிப்பது அதிக பயனுள்ளதாக இருக்கும். இத்தகைய சூழ்நிலையில், பாய்வுக் கட்டுப்பாடு நிரலின் ஒரு தொகுதியிலிருந்து மற்றொரு பகுதிக்கு தாவச் செய்கிறது. அத்தகைய தாவலுக்கு காரணமான நிரல் கூற்றுகள் கட்டுப்பாடு கூற்றுகள் எனப்படும்.

இந்த பாடப்பகுதியில் கட்டுப்பாட்டு கட்டமைப்பின் அடிப்படை செயல்பாடுகளான தேர்ந்தெடுப்பு, மடக்கு மற்றும் தாவுதல் கூற்றுகள் பற்றி காண்போம்.

10.2 கூற்றுகள் (Statements)

கணிப்பொறி நிரல் என்பது ஒரு குறிப்பிட்ட பணியை நிறைவேற்ற கொடுக்கப்படும் கூற்றுகள் அல்லது கட்டளைகளின் தொகுப்பாகும். இவ்வகை கூற்றுகள், ஒரு குறிப்பிட்ட செயலை நிறைவேற்ற பயன்படுகிறது. செயல் எனப்படுவது, ஒரு மாறி அறிவித்தல், கோவைகளின் மதிப்பீடு, மதிப்பிருத்தல், தீர்மானிப்பு, மடக்கு போன்ற செயல்பாடுகளாக இருக்கலாம்.

C++ நிரலில் இரு வகையான கூற்றுகள் பயன்படுத்தப்படுகின்றன.

(i) வெற்று கூற்று (Null Statements)

(ii) கலவை கூற்று (Compound Statements)

10.2.1 வெற்று கூற்று

அரைப்புள்ளியை மட்டுமே கொண்டிருக்கும் கூற்றை "வெற்று அல்லது வெறுமைக் கூற்று" எனப்படும். இதன் பொது வடிவம் பின்வருமாறு:

; // இது ஒரு வெற்றுக் கூற்று

வெற்றுக் கூற்றுகளை, மடக்கில் இட உணர்த்திகளாக பயன்படுத்தலாம் அல்லது

கலவை கூற்றுகள் அல்லது செயற்கூறுகளின் முடிவில் இடம் பெறச்செய்யலாம்.

10.2.2 கலவை (தொகுதி) கூற்று

C++ கூற்றுகளின் தொகுப்பினை நெளிவு அடைப்புக்குறிகளுக்குள் உள்ளடக்க அனுமதிக்கிறது. இந்த கூற்றுகளின் தொகுப்பினை கலவை கூற்று அல்லது தொகுதி என்கிறோம்.

கலவை கூற்றின் பொது வடிவம்

```
{
        கூற்று1;
        கூற்று 2;
        கூற்று 3;
}
```

எடுத்துக்காட்டு

```
{
        int x, y;
        x = 10;
        y = x + 10;
}
```

கலவை கூற்று அல்லது தொகுதி ஒற்றை அலகாக கருதப்படுகிறது. இவை நிரலின் எந்த பகுதியில் வேண்டுமானாலும் இடம் பெறலாம்.

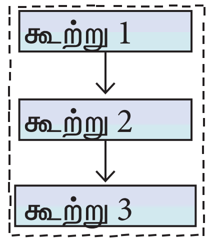
10.3 கட்டுப்பாட்டு கூற்றுகள்

கட்டுப்பாட்டு கூற்றுகள், கட்டளைகளின் பாய்வு வரிசை முறையை மாற்றி அமைக்கும். ஒரு நிரலிலுள்ள கூற்றுகள், வரிசைமுறை, தேர்ந்தெடுப்பு மற்றும் மடக்கு போன்ற கட்டுப்பாட்டு கூற்றுகளாக இயக்கப்படுகிறது. அனைத்து நிரலாக்க மொழிகளும், வரிசைமுறை, தேர்ந்தெடுப்பு மற்றும் மடக்கு கூற்றுகளை கொண்டுள்ளது.

கூற்றுகள் ஒன்றன் பின் ஒன்றாய் வரிசைமுறையில் நிறைவேற்றப்பட்டால், இந்த பாய்வை வரிசைமுறைப் பாய்வு என்கிறோம்.

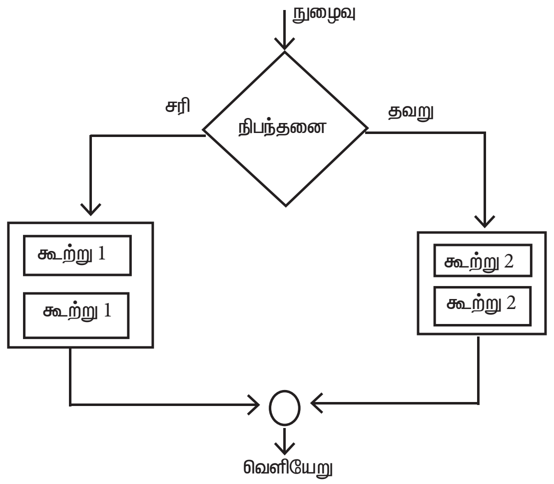
சில சூழ்நிலைகளில், கிளை பிரித்தல், மடக்கு, தாவுதல் மற்றும் செயற்கூறு அழைப்பு போன்ற கூற்றுகள் பாய்வின் ஓட்டத்தை மாற்றியமைக்கும். இந்த செயல்முறையை பாய்வுக் கட்டுப்பாடு (Control Flow) என்கிறோம்.

வரிசைமுறை கூற்று



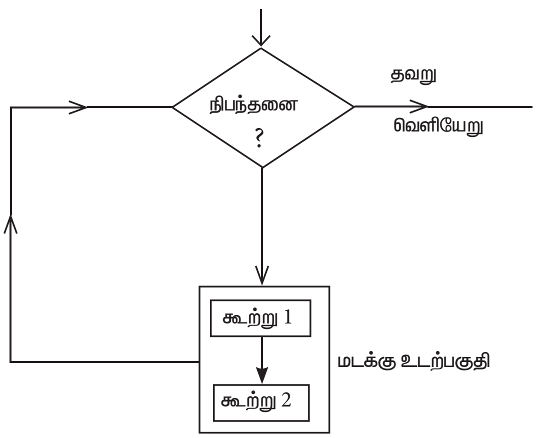
வரிசைமுறை கூற்றுகள் என்பது மேலிருந்து கீழாக ஒன்றன் பின் ஒன்றாக நிறைவேற்றப்படும் கூற்றுகளாகும். இத்தகைய கூற்றுகள் பாய்வு ஓட்டத்தை மாற்றி அமைக்காது. இவை எப்பொழுதும் அரைப்புள்ளியுடன் (;) முற்றுப்பெறுகிறது.

தேர்ந்தெடுப்புக் கூற்று



நிபந்தனை அடிப்படையில் நிறைவேற்றப்படும் கூற்றுகள் தேர்ந்தெடுப்பு கூற்றுகள் எனப்படும். கொடுக்கப்பட்ட நிபந்தனை சரி எனில் சரி கட்டளைத் தொகுதி (கூற்றுகளின் தொகுப்பு) இயக்கப்படும், இல்லையெனில் தவறு கட்டளைத் தொகுதி இயக்கப்படும். நிறைவேற்றப்பட வேண்டிய கூற்றுகளின் தொகுப்பை தீர்மானிக்க உதவுவதால், இக்கூற்றினை தீர்மானிப்புக் கூற்று அல்லது தேர்ந்தெடுப்பு கூற்று எனலாம்.

மடக்குக் கூற்று



மடக்குக் கூற்று என்பது ஒரு கட்டளைத் தொகுதியை நிபந்தனை அடிப்படையில் மீண்டும் மீண்டும் செயல்படுத்தும். கொடுக்கப்பட்ட நிபந்தனை சரி என இருக்கும் வரை, கட்டளைத் தொகுதி மீண்டும் மீண்டும் நிறைவேற்றப்படும். நிபந்தனை தவறாகும் போது தொடர்ந்து இயக்கப்படுவது நிறுத்தப்படுகிறது. இதனை மடக்குக் கூற்று அல்லது பன்முறைச் செயல் கூற்று என்கிறோம். மேலே கொடுக்கப்பட்டுள்ள பாய்வுப் படம் பன்முறைச் செயல் கூற்றினை விளக்குகிறது.

மீண்டும் மீண்டும் இயக்கப்படும் கட்டளைத் தொகுதியினை மடக்கின் உடற்பகுதி என்கிறோம். ஒரு மடக்கில் உள் நுழைய அல்லது வெளியேற காரணமான நிபந்தனையை வெளியேறல் நிபந்தனை அல்லது பரிசோதிப்பு நிபந்தனை என்கிறோம். பொதுவாக, அனைத்து வகை நிரலாக்க மொழிகளுக்கு சிக்கல்களின் தேவைக்கேற்ப இவ்வகை கூற்றினை நிரல் குறிமுறை எழுத பயன்படுத்திக் கொள்கிறது. C++ நிரலாக்க மொழியும் இவ்வகை கூற்றுகளை பயன்படுத்திக் கொள்கிறது. தேர்ந்தெடுப்புக் கூற்று மற்றும் மடக்குக் கூற்றுகள் கொடுக்கப்படும் நிபந்தனை கோவையின் வெளியீட்டு மதிப்பை அடிப்படையாக கொண்டு இயங்குகிறது. நிபந்தனை கோவை சரி அல்லது தவறு என்ற மதிப்புகளை விடையாக தரும்.

குறிப்பு

குறிப்பு:

C++ மொழியில், சுழியம் அல்லாத எந்தவொரு எண்ணும் "சரி" என எடுத்துக் கொள்ளப்படும், (எதிர்மறை எண் உள்பட) மற்றும் சுழியம் "தவறு" என்று எடுத்துக் கொள்ளப்படும்.

10.4 தேர்ந்தெடுப்பு கூற்றுகள்

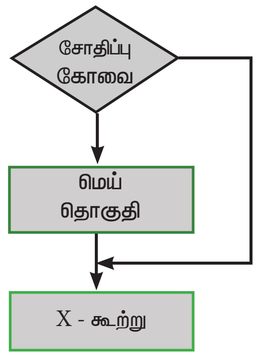
ஒரு தேர்ந்தெடுப்பு அமைப்பானது, நிபந்தனையின் அடிப்படையில் ஒரு கூற்றையோ அல்லது கட்டளைத் தொகுப்பையோ தேர்ந்தெடுத்து இயக்குகிறது. ஒரு குறிப்பிட்ட நிபந்தனை சரி எனில், ஒரு கூற்று அல்லது கட்டளைத் தொகுதி இயக்கப்படும், தவறு எனில் அவை இயக்கப்படாது. இதனை தேர்ந்தெடுப்பு கூற்று அல்லது தீர்மானிப்பு கூற்று என்கிறோம். C++ மொழியில் இரண்டு வகையான தேர்ந்தெடுப்பு கூற்றுகள் உள்ளன அவை, if மற்றும் switch கூற்றுகளாகும்.

10.4.1 if கூற்று

if கூற்று என்பது ஒரு தீர்மானிப்புக் கூற்றாகும். if கூற்று நிபந்தனையை சோதித்து, நிபந்தனை சரி எனில் சரித்தொகுதி கூற்று அல்லது கட்டளைத்தொகுதி இயக்கப்படும். இல்லையெனில் சரித் தொகுதி தவிர்க்கப்படும். if கூற்றின் கட்டளையமைப்பு.

```
if (நிபந்தனை கோவை)
        மெய் தொகுதி;
x-கூற்று
```

மேற்கண்ட கட்டளை அமைப்பில், if என்ற சிறப்புச்சொல் கோவை அல்லது நிபந்தனையை அடைப்புக்குறிக்குள் கொண்டிருக்க வேண்டும். கொடுக்கப்பட்ட கோவையின் மதிப்பு சரி எனில் (0 சுழியம் அல்லாத) சரித்தொகுதி இயக்கப்படும். அதைத் தொடர்ந்து x-கூற்று எடுத்துச் செல்லும். ஒரு சரித் தொகுதி ஒற்றை கூற்று, கலவை கூற்று அல்லது வெற்று கூற்றினை கொண்டிருக்கலாம்.



எடுத்துக்காட்டு 10.1 if கூற்றைப் பயன்படுத்தி மொத்த செலவினத்தை கணக்கிட C++ நிரல் ஒன்றை எழுதுக.

```
#include<iostream>
using namespace std;
int main()
{
        int qty,dis=0;
        float rate,tot;
        cout<<"\nEnter the quantity ";
        cin>>qty;
        cout<<"\nEnter the rate ";
        cin>>rate;
        if ( qty> 500)
        dis=10;
        tot = (qty * rate) - ( qty * rate * dis / 100);
        cout<<"The total expenses is "<< tot;
        return 0;
}
```

வெளியீடு

முதலாவது இயக்கம்

```
Enter the quantity 550
Enter the rate 10
The total expenses is 4950
```

இரண்டாவது இயக்கம்

```
Enter the quantity 450
Enter the rate 10
The total expenses is 4500
```

மேற்கண்ட நிரல் முதலில் இயக்கப்படும்போது, dis மற்றும் rate என்ற மாறிகளுக்கு முறையே, 550 மற்றும் 10 என்ற மதிப்புகள் உள்ளீடாக கொடுக்கப்பட்டுள்ளது. if கூற்றின் நிபந்தனை கோவை qty > 500 என வரைக்கப்பட்டுள்ளது. உள்ளீடாக பெறப்பட்ட qty மதிப்பு 550 என்பதன் அடிப்படையில், நிபந்தனை கோவையின் மதிப்பு சரி என அமையும். நிபந்தனை கோவை சரி என அமைவதால், நிரலின் தொடக்கத்தில் சுழியமாக இருந்த dis ன் மதிப்பு, தற்போது 10 என எடுத்துக்கொள்ளப்படும். இதனடிப்படையில், அடுத்த நிரல் கூற்றான tot = (qty * rate) - (qty * rate * dis / 100); என்பது இயக்கப்பட்டு, இறுதி மதிப்பு 4950 என்பது tot என்ற மாறியில் இருத்தப்படும்.

Chapter 10.indd 57 03/09/18 4:13 PM

மேற்கண்ட நிரல் இரண்டாவது முறையாக இயக்கப்படும்போது, dis மற்றும் rate என்ற மாறிகளுக்கு முறையே, 450 மற்றும் 10 என்ற மதிப்புகள் உள்ளீடாக கொடுக்கப்பட்டுள்ளது. if கூற்றின் நிபந்தனை கோவை qty > 500 என வரைக்கப்பட்டுள்ளது. உள்ளீடாக பெறப்பட்ட qty மதிப்பு 450 என்பதன் அடிப்படையில், நிபந்தனை கோவையின் மதிப்பு தவறு என அமையும். நிபந்தனை கோவை தவறு என அமைவதால், அதனைத் தொடர்ந்து வரும் dis = 10; என்ற மதிப்பிருத்துக் கூற்று, இயக்கப்படுவதில்லை. எனவே, நிரலின் தொடக்கத்தில் வரையறுக்கப்பட்ட dis = 0; என்பது எடுத்துக்கொள்ளப்பட்டு, நிரல் கூற்றான tot = (qty * rate) - (qty * rate * dis / 100); என்பது இயக்கப்படும். அதாவது, dis ன் மதிப்பு 0 என்றிருப்பதால், இந்த முறை தள்ளுபடி ஏதும் இல்லை. எனவே, tot மாறியின் இறுதி மதிப்பு 4500 என வரும்.

எடுத்துக்காட்டு 10.2 if கூற்றைப் பயன்படுத்தி ஒரு நபர் வாக்களிக்க தகுதியானவரா என சோதிக்க C++ நிரல் ஒன்று எழுதுக.

```
#include <iostream>
using namespace std;
int main()
{
        int age;
        cout << "\n Enter your age: ";
        cin >> age;
        if (age >= 18)
                cout<< "\n You are eligible for voting ....";
        cout << "This statement is always executed.";
        return 0;
}
```

if கூற்று, அதற்கு கீழே கொடுக்கப்படும் ஒரே ஒரு நிரல் கூற்றை மட்டுமே இயக்கும். எனவே, நெளிவு அடைப்புக் குறியீடு பயன்படுத்த வேண்டிய அவசியமில்லை.

வெளியீடு

```
Enter your age: 23
You are eligible for voting….
This statement is always executed.
```

எடுத்துக்காட்டு 10.3 if கூற்றைப் பயன்படுத்தி ஊக்க ஊதியம் (bonus) கணக்கீடு செய்ய ஒரு நிரல் எழுதுக.

```
#include<iostream>
using namespace std;
int main()
{
int bonus,yr_of_ser;
cout<<"\nEnter your year of service ";
cin>>yr_of_ser;
        if ( yr_of_ser> 3 )
        {
        bonus=2000;
        cout<<"\n Your bonus is " <<bonus;
```

if நிபந்தனைக் கூற்று, ஒன்றுக்கும் மேற்பட்ட நிரல் கூற்றுகளை இயக்க வேண்டுமெனில் அந்நிரல் கூற்றுகள் அனைத்தும் நெளிவு அடைப்புக் குறிக்குள் தரப்பட வேண்டும்.

Chapter 10.indd 58 03/09/18 4:13 PM

```
}
cout<<"\nCongratulations...";
return 0;
}
```

வெளியீடு

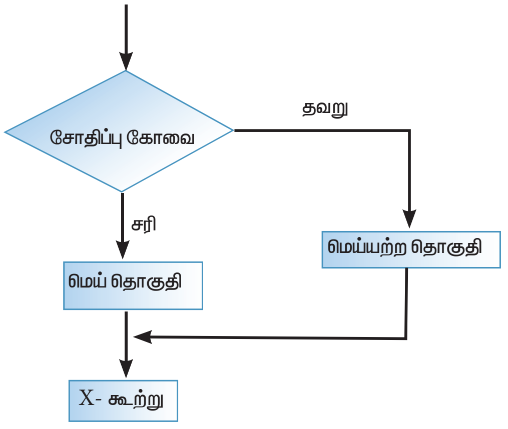
```
Enter your year of service 5
Your bonus is 2000
Congratulations...
```

10.4.2 if-else கூற்று

ஒரு if else கூற்று என்பதும் கட்டுப்பாட்டு பாய்வு கூற்றாகும். இது நிபந்தனையின் அடிப்படையில் ஒரு செயலை செய்யப் பயன்படுகிறது. If else கூற்றில், முதலில் கோவை அல்லது நிபந்தனை சரியா அல்லது தவறா என மதிப்பிடப்படுகிறது. கிடைக்கப்பெற்ற விடை சரியெனில், சரித்தொகுதியுனுள் இடம் பெறும் கூற்றுகள் இயக்கப்படும். தவறு தொகுதி நிறைவேற்றப்படாது, கிடைக்கபெற்ற விடை தவறு எனில், தவறு தொகுதியினுள் இடம் பெறும் கூற்றுகள் இயக்கப்படும், சரித்தொகுதி நிறைவேற்றப்படாது. If else கூற்றின் கட்டளை அமைப்பு பின்வருமாறு:

```
if (கோவை)
{
        மெய் தொகுதி;
}
else
{
        மெய்யற்ற தொகுதி;
}
x-கூற்று
```

மேற்கண்ட கட்டளை அமைப்பில், if கூற்றுடன் கொடுக்கப்பட்ட நிபந்தனை முதலில் மதிப்பீடு செய்யப்படுகிறது. நிபந்தனை சரி எனில், சரித்தொகுதி இயக்கப்பட்டு, பின் கட்டுப்பாடு X கூற்றுக்கு இட்டுச் செல்கிறது,. நிபந்தனை தவறு எனில், தவறு தொகுதி இயக்கப்பட்டு, பின் கட்டப்பாடு X கூற்றுக்கு அனுப்பிவைக்கப்படுகிறது.



Chapter 10.indd 59
03/09/18 4:13 PM

எடுத்துக்காட்டு 10.4 if-else கூற்றைப் பயன்படுத்தி கொடுக்கப்பட்ட எண் ஒற்றைப் படை அல்லது இரட்டைப் படை எண்ணா எனக் காணும் C++ நிரல் ஒன்றை எழுதுக.

```
#include <iostream>
using namespace std;
int main()
{
        int num, rem;
        cout<< "\n Enter a number: ";
        cin>>num;
        rem = num % 2;
        if (rem==0)
                cout<< "\n The given number" <<num<< " is Even";
        else
                cout<< "\n The given number "<<num<< " is Odd";
        return 0;
}
```

வெளியீடு

```
        Enter number: 10
        The given number 10 is Even
```

மேற்கண்ட நிரலில், கொடுக்கப்பட்ட எண்ணின் வகுமீதி rem என்ற மாறியில் இருத்தப்படும். rem ன் மதிப்பு சுழியம் எனில், கொடுக்கப்பட்ட எண் இரட்டைப் படை எண் (even number) என்ற வெளியீடும், இல்லையெனில், ஒற்றைப் படைஎண் (odd number) என்ற வெளியீடும் கிடைக்கும்.

10.4.3 பின்னலான if கூற்று:

ஒரு if கூற்று, மற்றொரு if கூற்றை கொண்டிருந்தால் அது பின்னலான if கூற்று என்று அழைக்கப்படுகிறது. பின்னலான if கூற்றுகள், பின்வரும் மூன்று வடிவங்களில் அமைக்கப்படும். அவையாவன

(1) ஒரு if கூற்றுக்கு உள்ளே மற்றொரு if கூற்றை அமைத்தல்

(2) ஒரு if கூற்றின் else கூற்றுக்குள் மற்றொரு if கூற்றை அமைத்தல்

(3) ஒரு if கூற்றுக்கு உள்ளே ஒரு if கூற்றும், else கூற்றுக்கு உள்ளே மற்றொரு if கூற்றும் அமைத்தல்.

இந்த மூன்று வடிவங்களின் கட்டளை அமைப்பு கீழே கொடுக்கப்பட்டுள்ளது.

Chapter 10.indd 60 03/09/18 4:13 PM

ஒரு if கூற்றுக்கு உள்ளே மற்றொரு if கூற்றை அமைத்தல்

```
if (கோவை-1)
{
        if (கோவை)
        {
          மெய் பகுதி கூற்றுகள்;
        }
        else
        {
         மெய்யற்ற பகுதி கூற்றுகள்;
        }
}
else
    else பகுதியின் உடற்பகுதி;
```

ஒரு if கூற்றின் else கூற்றுக்குள் மற்றொரு if கூற்றை அமைத்தல்

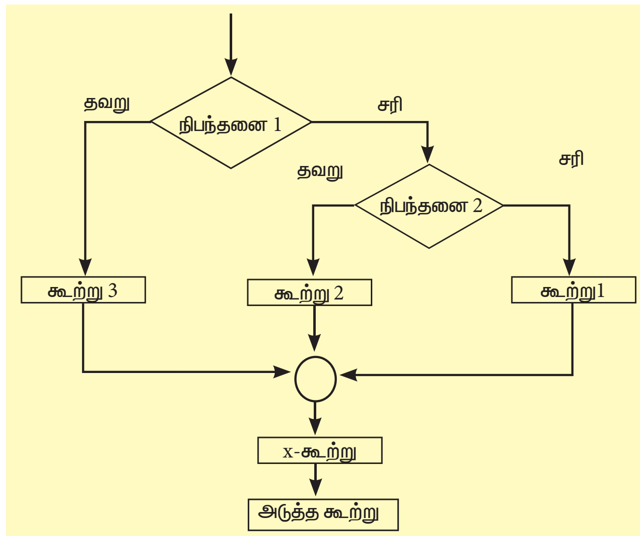
```
if (கோவை-1)
{
        மெய் பகுதியின் உடற்பகுதி;
}
else
{
        if (கோவை)
        {
          மெய் பகுதி கூற்றுகள்;
        }
        else
        {
         மெய்யற்ற பகுதி கூற்றுகள்;
        }
}
```

ஒரு if கூற்றுக்கு உள்ளே ஒரு if கூற்றும், else கூற்றுக்கு உள்ளே மற்றொரு if கூற்றும் அமைத்தல்.

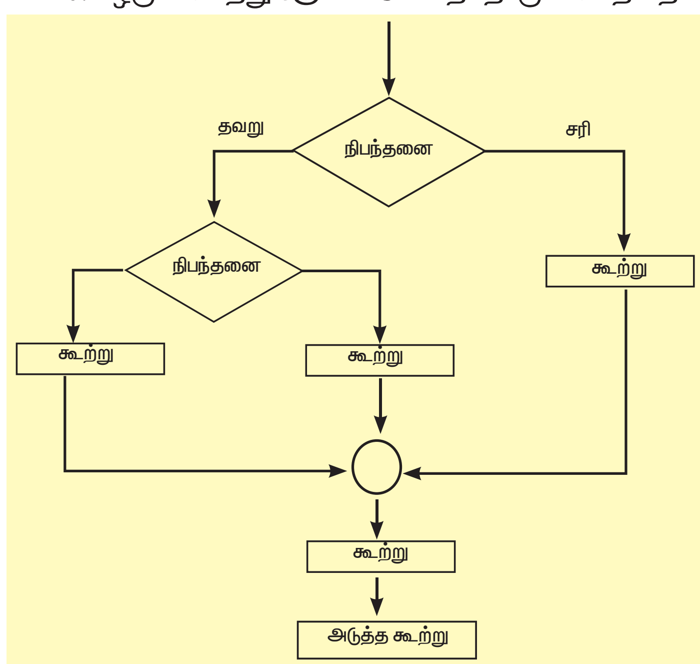
```
if (கோவை)
{
        if (கோவை)
        {
          மெய் பகுதி கூற்றுகள்;
        }
        else
        {
          மெய்யற்ற பகுதி கூற்றுகள்;
        }
}
else
{
        if (கோவை)
        {
          மெய் பகுதி கூற்றுகள்;
        }
        else
        {
          மெய்யற்ற பகுதி கூற்றுகள்;
        }
}
```

மேலே கொடுக்கப்பட்டுள்ள முதலாவது கட்டளை அமைப்பில், முதன்மை if கூற்றுக்குள் கொடுக்கப்பட்டுள்ள கோவை-1 முதலில் மதிப்பிடப்படும். அதன் மதிப்பு சரி எனில், அதனுள் கொடுக்கப்பட்டுள்ள பின்னலான if கூற்றுக்கு கட்டுபாட்டு பாய்வு செல்லும். பின்னலான if கூற்றின் நிபந்தனை சரி எனின், அதனுள் கொடுக்கப்பட்ட சரி தொகுதியின் கூற்றும், தவறு எனில் else கூற்றினுள் கொடுக்கப்பட்டுள்ள தவறு தொகுதியின் கூற்றும் இயக்கப்படும். முதன்மை if கூற்றின் நிபந்தனை கோவை மதிப்பு தவறு எனில், முதன்மை else தொகுதிக்குள் கொடுக்கப்பட்டுள்ள நிரல் கூற்றுகள் இயங்கும்.

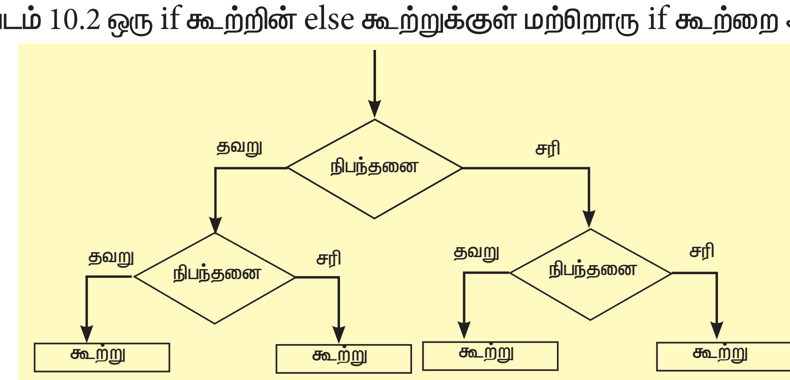
மேலே கொடுக்கப்பட்டுள்ள மூன்று வகையான பின்னல் if கூற்றுகளின் பாய்வுப் படங்கள் கீழே கொடுக்கப்பட்டுள்ளது.



பாய்வுப் படம் 10.1 ஒரு if கூற்றுக்கு உள்ளே மற்றொரு if கூற்றை அமைத்தல்



பாய்வுப் படம் 10.2 ஒரு if கூற்றின் else கூற்றுக்குள் மற்றொரு if கூற்றை அமைத்தல்



பாய்வுப் படம் 10.3 ஒரு if கூற்றுக்கு உள்ளே ஒரு if கூற்றும், else கூற்றுக்கு உள்ளே மற்றொரு if கூற்றும் அமைத்தல்.

Chapter 10.indd 62 03/09/18 4:13 PM

எடுத்துக்காட்டு **10.5** பின்னலான **if** கூற்றினை பயன்படுத்தி தரநிலைக்கு ஏற்ப விற்பனை தரகை **(Commission)** கணக்கிட நிரல் ஒன்று எழுது.

```
#include <iostream>
using namespace std;
int main()
{
        int sales, commission;
        char grade;
        cout << "\n Enter Sales amount: ";
        cin >> sales;
        cout << "\n Enter Grade: ";
        cin >> grade;
        if (sales > 5000)
        {
                commission = sales * 0.10;
                cout << "\n Commission: " << commission;
        }
        else
        {
                commission = sales * 0.05;
                cout << "\n Commission: " << commission;
        }
        cout << "\n Good Job ..... ";
        return 0;
}
```

வெளியீடு:

```
        Enter Sales amount: 6000
        Enter Grade: A
        Commission: 600
        Good Job .....
```

10.4.4 if -else-if அடுக்கு

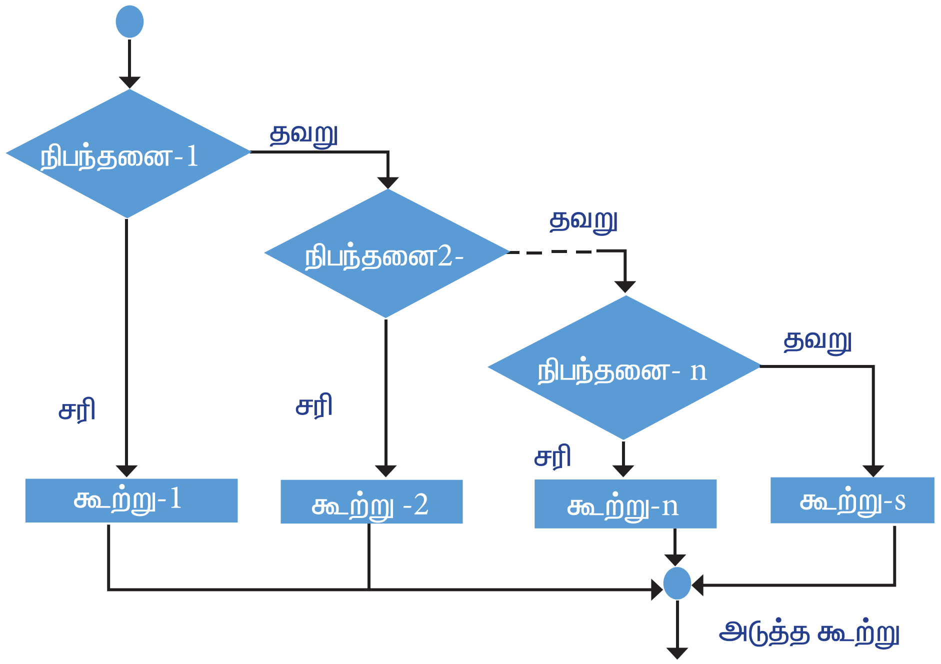
if-else அடுக்கு என்பது பல வழி தீர்மானிப்பு கூற்றாகும். இந்த வகையான கூற்றில், if என்னும் சிறப்புச் சொல்லைத் தொடர்ந்து ஒன்று அல்லது ஒன்றுக்கு மேற்பட்ட else if கூற்றுகள் இடம் பெற்றிருக்கும். இறுதியாக else கூற்றுடன் முடிவு பெறும்.

if-else அடுக்கின் கட்டளை அமைப்பு:

Chapter 10.indd 63 03/09/18 4:13 PM

```
if (கோவை-1)
{
  கூற்று-1
}
else
          if(கோவை- 2)
          {
                    கூற்று-2
          }
          else
                    if (கோவை-3)
                    {
                              கூற்று-3
                    }
                    else
                    {
                              கூற்று-4
                    }
```

கொடுக்கப்பட்டுள்ள கோவைகளில் (Expressions) சரி எனில், அந்த தொகுதியுடன் தொடர்புடைய கூற்று நிறைவேற்றப்படும், அந்த அடுக்கில் இடம் பெற்றிருக்கும் மற்ற கூற்றுகள் தவிர்க்கப்பட்டு விடும்.



பாய்வுப் படம் 10.4 if-else அடுக்கு

எடுத்துக்காட்டு 10.6 if-else அடுக்கினை பயன்படுத்தி உன் தரநிலைக் கண்டறியும் C++ நிரல் எழுதுக.

```
#include <iostream>
using namespace std;
int main ()
{
int marks;
cout<<" Enter the Marks  :";
cin>>marks;
if( marks >= 60 )
        cout<< "Your grade is 1st class !!" <<endl;
                else if( marks >= 50 && marks < 60)
                        cout<< "your grade is 2nd class !!" <<endl;
                                else if( marks >= 40 && marks < 50)
                                        cout<< "your grade is 3rd class !!" <<endl;
else
        cout<< "You are fail !!" <<endl;
return 0;
}
```

வெளியீடு

```
        Enter the Marks  :60
        Your grade is 1st class !!
```

மதிப்பெண்கள் 60ஐ விட அதிகமானதாகவோ அல்லது சமமானதாகவோ இருப்பின் "Your grade is 1st class !!" என்ற செய்தி வெளியிடப்படும். அடுக்கில் இடம் பெற்ற மற்றவை தவிர்க்கப்பட்டு விடும். மதிப்பெண்கள் 50க்கும், 59க்கும் இடையே இருக்குமானால் "Your grade is 2nd class !!" என்று வெளியிடப்படும், மற்ற அடுக்கு தவிர்க்கப்பட்டு விடும். மதிப்பெண்கள் 40 க்கும் 49க்கும் இடையில் இருந்தால் செய்தி "Your grade is 3rd class !!" என்று வெளியிடப்படும் இல்லையெனில் "You are fail !!" என்ற செய்தி காண்பிக்கப்படும்.

10.4.5 if else க்கு மாற்றான நிபந்தனை செயற்குறி

நிபந்தனை செயற்குறி அல்லது மும்ம செயற்குறி என்பது if else கூற்றின் மாற்று வழியாகும். நிபந்தனை செயற்குறி ?: என்ற இரண்டு குறியீடுகளை கொண்டுள்ளது. இது மூன்று செயலுருப்புகளை ஏற்கும். நிபந்தனை செயற்குறியின் பாய்வுக்கட்டுப்பாடு கீழே கொடுக்கப்பட்டுள்ளது,

நிபந்தனை செயற்குறியின் கட்டளை அமைப்பு

கோவை 1 ? கோவை 2 : கோவை 3 ;

மேற்கண்ட கட்டளை அமைப்பில், கோவை 1 என்பது மதிப்பிட வேண்டிய நிபந்தனையை குறிக்கும். நிபந்தனை சரி எனில், கட்டுப்பாடு கோவை 2க்குச் செல்லும். இல்லையெனில் கட்டுப்பாடு கோவை 3க்கு எடுத்துச்செல்லப்படும்.

Chapter 10.indd 65 03/09/18 4:13 PM

எடுத்துக்காட்டு 10.7 கொடுக்கப்பட்ட இரண்டு எண்களில் எது பெரியது என கண்டறிய நிபந்தனை செயற்குறியை பயன்படுத்தி ஒரு நிரல் எழுதுக.

```
#include <iostream>
using namespace std;
int main()
{
        int a, b, largest;
        cout << "\n Enter any two numbers: ";
        cin >> a >> b;
         largest = (a>b)? a : b;
        cout << "\n Largest number : " << largest;
        return 0;
}
```

வெளியீடு:

```
        Enter any two numbers: 12 98
        Largest number : 98
```

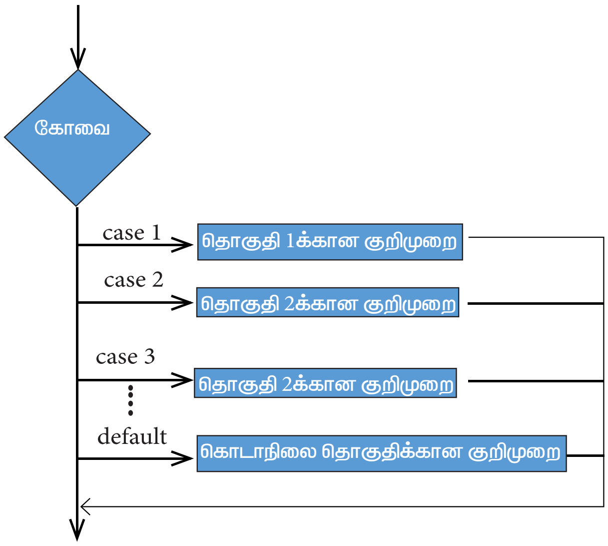
10.4.6 switch கூற்று

switch கூற்று என்பது ஒரு பல வழி கிளைப்பிரிப்பு கூற்றாகும். இது கோவையின் மதிப்பின் அடிப்படையில் பல்வேறு நிரல் பகுதிகளுக்கு கட்டுப்பாட்டை எடுத்துச் செல்வதற்கு எளிதாக வகை செய்கிறது. switch கூற்று ஒன்றுக்கும் மேற்ப்பட்ட if else கூற்றுகளுக்கு மாற்றாக அமைந்துள்ளது.

witch கூற்றின் கட்டளை அமைப்பு;

```
switch(கோவை)
{
        case constant 1:
                கூற்று(s);
                break;
        case constant 2:
                கூற்று(s);
                break;
        .
        .
        .
        .
        default:
                கூற்று (s);
}
```

மேற்கண்ட கட்டளை அமைப்பில், கோவை மதிப்பீடு செய்யப்பட்டு, பெறப்பட்ட மதிப்பு ஏதேனும் ஒரு case கூற்றில் கொடுக்கப்பட்ட மாறிலியின் மதிப்புடன் பொருந்தியிருப்பின், அதனுடன் தொடர்புடைய கட்டளைத் தொகுப்பு நிறைவேற்றப்படும். switch கூற்றின் இயங்கு முறை பாய்வுப்படமும் படம் 10.5ல் கொடுக்கப்பட்டுள்ளது.



படம் 10.5 switch கூற்றின் இயங்குமுறை

விதிமுறைகள்:

1. switch கூற்றில் தரப்பட்டுள்ள கோவையின் விடை மாறிலி மதிப்பாக இருத்தல் வேண்டும் இல்லையெனில அது ஒரு பிழையான வடிவமாகும்.

Chapter 10.indd 66 03/09/18 4:13 PM

2. ஒரே மதிப்பு இரண்டு case கூற்றுகளில் அனுமதிக்கப்பட மாட்டாது.

3. default கூற்று கட்டாயமில்லை.

4. switch கூற்றின் நிரல் தொகுதியை நிறைவு செய்ய break கூற்று பயன்படுகிறது. நிரல் பாய்வு break கூற்றை அடையும் போது, switch கூற்றிலிருந்து வெளியேறி, switch கூற்றை தொடர்ந்து கொடுக்கப்பட்டுள்ள நிரல் கூற்றுகளை இயக்கும்.

5. break கூற்று கொடுப்பது கட்டாயமில்லை. break கூற்று கொடுக்கப்படவில்லை எனில், case நிபந்தனை நிறைவேற்றப்படும் நிரல் கூற்றை தொடர்ந்து உள்ள அனைத்து கூற்றுகளும் கட்டுப்பாடின்றி இயங்கும்.

6. பின்னலான switch கூற்றுகளும் அமைக்கப்படும்.

எடுத்துக்காட்டு 10.8 – switch கூற்றை விளக்கும் C++ நிரல்

```
#include <iostream>
using namespace std;
int main()
{
        int num;
        cout << "\n Enter week day number: ";
        cin >> num;
        switch (num)
        {
                case 1 : cout << "\n Sunday"; break;
                case 2 : cout << "\n Monday"; break;
                case 3 : cout << "\n Tuesday"; break;
                case 4 : cout << "\n Wednessday"; break;
                case 5 : cout << "\n Thursday"; break;
                case 6 : cout << "\n Friday"; break;
                case 7 : cout << "\n Saturday"; break;
                default: cout << "\n Wrong input....";
        }
}
```

வெளியீடு:

```
        Enter week day number: 6
        Friday
```


Chapter 10.indd 67 03/09/18 4:13 PM

எடுத்துக்காட்டு 10.9 – switch கூற்றை விளக்கும் C++ நிரல்

```
#include <iostream>
using namespace std;
int main()
{
        char grade;
        cout << "\n Enter Grade: ";
        cin >> grade;
        switch(grade)
        {
                case 'A' : cout << "\n Excellent...";
                break;
                case 'B' :
                case 'C' : cout << "\n Welldone ...";
                break;
                case 'D' : cout << "\n You passed ...";
                break;
                case 'E' : cout << "\n Better try again ...";
                break;
                default : cout << "\n Invalid Grade ...";
        }
cout << "\n Your grade is " << grade;
        return 0;
}
```

வெளியீடு:

```
        Enter Grade: C
        Welldone ...
        Your grade is C
```

10.4.7 switch எதிர் if-else : சில வேறுபாடுகள்

"if-else" மற்றும் "switch" ஆகிய இரண்டும் தேர்ந்தெடுப்பு கூற்றுகள். தேர்ந்தெடுப்பு கூற்றுகள், நிபந்தனை சரி அல்லது தவறு என்ற அடிப்படையில், நிரலின் கட்டுப்பாட்டை குறிப்பிட்ட கட்டளைத் தொகுதிக்கு எடுத்துச் செல்கிறது. எனினும், அவை செயல்படும் முறைகளில் சில வேறுபாடுகள் உள்ளன. அவை பின்வருமாறு

if-else மற்றும் switch கூற்றின் இடையே உள்ள முக்கிய வேறுபாடுகள்

1. if கூற்றின் உள்ளே அமைந்திருக்கும் கோவை, if தொகுதியின் உள்ளே தரப்பட்டுள்ள கட்டளைகளை செயல்படுத்துவதோ அல்லது else தொகுதியின் கீழ் உள்ளவையா என்பதை தீர்மானிக்க. அவ்வாறின்றி, Switch கூற்றினுள் அமைந்திருக்கும் கோவை,

2. ஒரு if else கூற்றானது, பல தேர்வுகளுக்கு பல கூற்றுகளை பயன்படுத்துகிறது. அவ்வாறின்றி, Switch கூற்றானது, பல தேர்வுகளுக்கு ஒரு கோவையைப் பயன்படுத்துகிறது.

3. if-else கூற்று சமமானதா என்பதையும் தருக்க (நிபந்தனை) கோவையையும் சரிபார்க்கிறது. ஆனால் Switch சம்மானதா என்பதை மட்டும் சரிபார்க்கிறது.

4. If-கூற்று முழு எண், குறியுரு, சுட்டு அல்லது மிதப்புப் புள்ளி வகை அல்லது பூலியன் வகை தரவினங்களை மதிப்பீடு செய்கிறது.

Chapter 10.indd 68 03/09/18 4:13 PM

அவ்வாறின்றி, switch கூற்று குறியுரு அல்லது முழு எண் தரவு வகையை மட்டுமே மதிப்பீடு செய்யும்.

5. if தொகுதியின் கீழுள்ள கட்டளைத் தொகுதி நிறைவேற்றப்படும் அல்லது else தொகுதியின் கீழுள்ள கட்டளைத் தொகுதி நிறைவேற்றப்படும். switch கூற்றினுள் அமைந்திருக்கும் கோவை தவறு என ஆகும் போது, default கூற்று இயக்கப்படும்.

6. if else கூற்றில் தேவையான திருத்தங்களை மேற்கொள்வதும், திருத்தம் தேவைப்படும் இடங்களை கண்டறிவதும் மிகவும் கடினம். எனவே if கூற்றை பதிப்பாய்வு செய்வது கடினமானதாகும், அவ்வாறின்றி, switch கூற்றுகளில் திருத்தங்களை எளிதாக கண்டறிய முடிவதால், பதிப்பாய்வு செய்வதும் எளிதாகிறது.

> switch கூற்றை விட if கூற்று மிகுந்த ஆற்றல் மிக்கது

switch கூற்றினைப் பற்றி தெரிந்துக் கொள்ள வேண்டிய சில முக்கிய குறிப்புகள்:

1. switch கூற்று, தரத்தை ஒப்பிடுதலைக்கான பணிக்கு மட்டுமே பயன்படுகிறது .

2. switch கூற்றில் இரு வெவ்வேறு case கூற்றுகளுக்கு ஒரே மதிப்புகள் இருக்கக் கூடாது.

3. switch கூற்றில் குறியுறு மாறிலிகள் பயன்படுத்தப்பட்டால், அவை தாமாகவே ASCII குறியீட்டின் சமமான மதிப்பிற்கு மாறிக் கொள்ளும்.

4. பல தேர்வுகளிலிருந்து ஒன்றை தேர்வு செய்ய வேண்டிய சூழ்நிலையில், switch கூற்று மிகவும் பொருத்தமான தேர்வு.

10.4.8 பின்னலான switch கூற்று

ஒரு switch கூற்று மற்றொரு switch கூற்றின் வரிசைமுறையின் ஒரு பகுதியாக அமைந்தால் அதனை பின்னலான switch கூற்று என்கிறோம். உள் switch கூற்று மற்றும் வெளி switch கூற்றின் மாறிலி ஒன்றாகவோ அல்லது வெவ்வேறாகவோ இருக்கலாம்.

பின்னலான switch கூற்றின் கட்டளை அமைப்பு;

```
switch (கோவை)
{
        case 1:
        கூற்று(s);
        break;
                switch (கோவை)
                {
                case1:
                கூற்று/கூற்றுகள்;
                        break;
                case 2:
                கூற்று /கூற்றுகள்;
                        break;
                        .
                        .
                        .
                default :
                கூற்று/கூற்றுகள்;
                }
case 2:
கூற்று/கூற்றுகள்;
break;
        .
        .
        .
default :
கூற்று/கூற்றுகள்;
}
```

பின்வரும் எடுத்துக்காட்டு நிரல் பின்னலான switch கூற்றினை விளக்குகிறது. வெளி switch கூற்று சுழியம் அல்லது சுழியம் அல்லாததை சோதிக்கும். உள் switch கூற்று ஒற்றைப்படை அல்லது இரட்டைப்படையை சோதிக்கும்.

Chapter 10.indd 69 03/09/18 4:13 PM

எடுத்துக்காட்டு 10.10 switch கூற்றை பயன்படுத்தி சுழியம் அல்லது சுழியம் அல்லாததையும் ஒற்றைப்படையா, இரட்டைப்படையா என்பதையும் கண்டறிய C++ நிரல் எழுதுக.

```
#include <iostream>
using namespace std;
int main()
{
int a = 8;
cout<<"The Number is : " <<a <<endl;
switch (a)
        {
        case 0 :
        cout<<"The number is zero" <<endl;
        break;
        default:
        cout<<"The number is a non-zero integer" <<endl;
        int b = a % 2;
        switch (b)
{
        case 0:
        cout<<"The number is even" <<endl;
        break;
        case 1:
        cout<<"The number is odd" <<endl;
        break;
        }
}
        return 0;
}
```

வெளியீடு

```
The Number is : 8
The number is a non-zero integer
The number is even
```

10.5 பன்முறைச் செயல் (அல்லது) மடக்கு

பன்முறைச் செயல் அல்லது மடக்கு என்பது ஒன்று அல்லது ஒன்றுக்கு மேற்பட்ட கூற்றுகள், நிபந்தனை நிறைவேறும் வரை திரும்பத் திரும்ப இயக்கப்படுவதை குறிக்கும். இந்த கூற்றுகளை பாய்வுக் கட்டுப்பாட்டு கூற்றுகள் எனவும் கூறலாம். இது நிரலின் நீளத்தை குறைக்கவும், நேரத்தை குறைக்கவும், நிரல் இயக்கவும், குறைவான நினைவக இடத்தை எடுத்துக் கொள்ளவும் பயன்படுகிறது. C++ மூன்று வகையான மடக்கு கூற்றுகளை ஏற்கிறது.

- for கூற்று
- while கூற்று
- do-while கூற்று

அனைத்து மடக்கு கூற்றுகளும் குறிப்பிட்ட நிபந்தனை சரி என இருக்கும் வரையில்

Chapter 10.indd 70 03/09/18 4:13 PM

கட்டளைத் தொகுதியை மீண்டும் மீண்டும் நிறைவேற்றுகிறது. குறிப்பிட்ட நிபந்தனை மடக்கு கட்டுப்பாடு எனப்படுகின்றது. மூன்று மடக்கு கூற்றுகளுக்கும், சுழியம் அல்லாத எந்தவோர் எண்ணும் சரி என எடுத்துக் கொள்ளப்படும் மற்றும் சுழியம் என்பது தவறு என எடுத்துக்கொள்ளலாம்.

மடக்கின் பகுதிகள்

ஒவ்வொரு மடக்கும் பல்வேறு பயன்பாடுகளுக்கான நான்கு கூறுகளை கொண்டுள்ளது.

- தொடக்க மதிப்பிருத்தும் கோவை
- சோதிப்பு நிபந்தனை கோவை
- புதுப்பித்தல் / மிகுத்தல் கோவை
- மடக்கின் உடற்பகுதி

தொடக்க மதிப்பிருத்தும் கோவைகள் : நிரலின் கட்டுப்பாடு மடக்கினுள் நுழையும் முன் கட்டுப்பாட்டு மாறிகள் தொடக்க மதிப்பிருத்தல் வேண்டும். ஒரு கட்டுப்பாட்டு மாறிக்கு தொடக்க மதிப்பிருத்தும் செயல்பாடு தொடக்க மதிப்பிருத்தும் கோவையின் கீழ் இடம் பெறும். கட்டுப்பாட்டு மாறியில் தொடக்க மதிப்பிருத்தல் மடக்கு செயல்படும் முன்பாக, ஒரே ஒரு முறை தான் செயல்படுத்தப்படும்.

பரிசோதிப்பு நிபந்தனை கோவை: பரிசோதிப்பு நிபந்தனை கோவை என்பது ஒரு மடக்கின் உள்ளே அமைந்துள்ள உடற்பகுதியிலிருக்கும் கட்டளைத் தொகுதி நிறைவேற்றப்பட வேண்டுமா இல்லையா என்பதை தீர்மானிக்கும் கோவை அல்லது நிபந்தனையாகும்.

கொடுக்கப்பட்ட நிபந்தனை சரி (அதாவது 1) எனில், மடக்கின் உடற்பகுதி செயல்படுத்தப்படும் இல்லையேல், மடக்கை விட்டு வெளியேறும்.

பரிசோதிப்பு நிபந்தனை கோவையின் சோதிப்பு நிலையை அடிப்படையாகக் கொண்டு, மடக்குகளை நுழைவு சோதிப்பு மடக்கு மற்றும் வெளியேறல் சோதிப்பு மடக்கு என இரண்டு வகையாக பிரிக்கப்படுகிறது.

நுழைவு சோதிப்பு மடக்கில், நிபந்தனை கோவை மடக்கினுள் நுழையும் முன் சோதிக்கப்படுகிறது. அவ்வாறின்றி வெளியேறல் சோதிப்பு மடக்கில் நிபந்தனைக் கோவை மடக்கினை விட்டு வெளியேறும் முன்பாக சோதிக்கப்படுகிறது.

புதுபித்தல் / மிகுத்தல் கோவை: இது மடக்கு கட்டுப்பாட்டு மாறியின் மதிப்பை மாற்றம் செய்வதற்கு பயன்படுகிறது. இந்த கூற்று, மடக்கின் உடற்பகுதி செயல்பட்டு முடிந்த பின், மடக்கின் இறுதியில் செயல்படுத்தப்படுகிறது.

மடக்கின் உடற்பகுதி: மடக்கின் உடற்பகுதி என்பது மீண்டும் மீண்டும் நிறைவேற்றப்பட வேண்டிய கூற்று அல்லது கட்டளைத் தொகுதியை குறிக்கும். நுழைவு சோதிப்பு மடக்கில், முதலில் நிபந்தனை கோவை செயல்படுத்தப்பட்டு, சுழியம் அல்லாத எண்ணாக இருப்பின், மடக்கின் உடற்பகுதி செயல்படுத்தப்படும் இல்லையெனில் மடக்கு முடிவு பெறும். வெளியேறல் சோதிப்பு மடக்கில், முதலில் மடக்கின் உடற்பகுதி செயல்படுத்தப்பட்டு, பின் நிபந்தனைக் கோவை சரிபார்க்கப்படுகிறது. சோதிப்புக் கோவை சரி எனில், மடக்கின் உடற்பகுதி திரும்ப நிறைவேற்றப்படும் இல்லையெனில் மடக்கு முடிவுப் பெறும்.

10.5.1 for மடக்கு

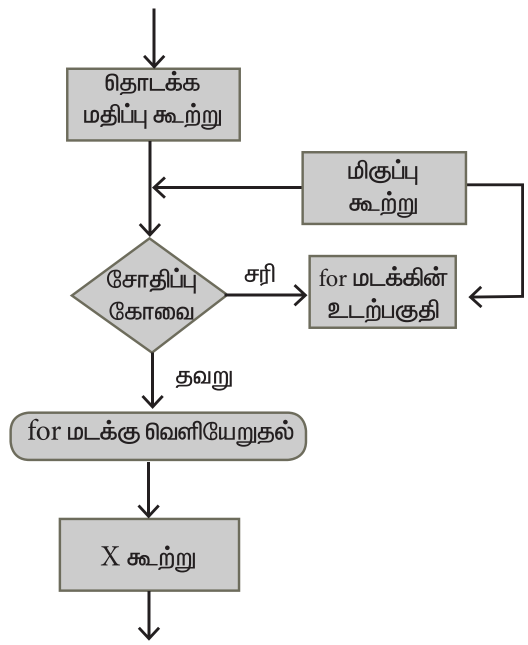
for மடக்கு ஓர் எளிய மடக்காகும். இது கூற்றுகளை மீண்டும் மீண்டும் இயக்கும். for மடக்கு மூன்று கூற்றுகளை கொண்டிருக்கும் அவை 1. தொடக்க மதிப்பிருத்தல் (Initialization), 2. சோதிப்பு நிபந்தனை அல்லது நிபந்தனை கோவை (Test-expression) மற்றும் 3. மிகுப்பு கோவைகள் (Update expressions). இவை அரைப்புள்ளியில் பிரிக்கப்பட்டிருக்க வேண்டும்.

பொது வடிவம்:

```
for (தொடக்க மதிப்பிருத்தல்; சோதிப்பு நிபந்தனை; மிகுப்பு கோவைகள்)
{
        நிரல் கூற்று 1;
        நிரல் கூற்று 2;
        .............
}
நிரல் கூற்று -x;
```

தொடக்க மதிப்பிருத்தல் பகுதி, கட்டுப்பாட்டு மாறியை அறிவிக்க அல்லது மாறியின் மதிப்பிருத்த பயன்படுகிறது. இந்த பகுதி ஒரு முறை மட்டும் இயக்கப்படும். பின்னர் நிரல் பாய்வு, சோதிப்பு நிபந்தனை கோவைக்கு செல்லும். கட்டுப்பாட்டு கோவையின் நிபந்தனை சோதிக்கப்பட்டு, அதன் மதிப்பு தவறு என்று வந்தால் நிரல் கூற்று-x இயங்கும். மதிப்பு சரியென்று வந்தால் for மடக்கின் உடற்பகுதி இயக்கப்பட்டு, பின்னர் நிரல் பாய்வு மிகுப்பு கோவைக்கு அனுப்பி வைக்கப்படும். மிகுப்பு கோவை செயல்பட்ட பின்னர், மீண்டும் சோதிப்பு நிபந்தனை கோவைக்கு அனுப்பி வைக்கப்படும்.

for மடக்கு இயக்கும் முறை மற்றும் பாய்வு படம் 10.6 ல் விளக்கப்பட்டுள்ளது.



படம் 10.6: for மடக்கின் பாய்வு

எடுத்துக்காட்டு 10.11: சுழியம் (0) முதல் ஒன்பது (9) வரை உள்ள எண்களை வெளியிட for மடக்கை பயன்படுத்தி நிரல் ஒன்று எழுதுக.

```
#include <iostream>
using namespace std;
int main ()
{
int i;
for(i = 0; i < 10; i ++ )
        cout<< "value of i : " <<i<<endl;
return 0;
}
```


Chapter 10.indd 72 03/09/18 4:13 PM

வெளியீடு

```
value of i : 0
value of i : 1
value of i : 2
value of i : 3
value of i : 4
value of i : 5
value of i : 6
value of i : 7
value of i : 8
value of i : 9
```

மேலே கொடுக்கப்பட்டுள்ள நிரலில் for மடக்கு இயங்கும் முறை கீழே கொடுக்கப்பட்டுள்ளது.

சோதிப்பு நிபந்தனை   மிகுப்பு கோவைகள்

தொடக்க மதிப்பிருத்தல்

for (i=0;   i < 10;   i++)

cout<<"value of i:"<<i<<endl;

மடக்கின் உடற்பகுதி

இங்கே மடக்கின் உடற்பகுதியில் ஒரே ஒரு நிரல்கூற்று மட்டுமே உள்ளது. எனவே அதனை நெளிவு அடைப்புக் குறிக்குள் கொடுக்க வேண்டிய அவசியமில்லை.

மேலே கொடுக்கப்பட்டுள்ள நிரலில், முதலில் கட்டுப்பாட்டு மாறி i க்கு தொடக்க மதிப்பிருத்தப்படும், பின்னர் i மாறியின் தொடக்க மதிப்பு 10 என்ற எண்ணுடன் ஒப்பிடப்படும். ஒப்பீட்டின் முடிவு சரி என இருந்தால், நிரல் கட்டுப்பாட்டு பாய்வு, இயக்கப்பட வேண்டிய நிரல் கூற்று cout<< "value of i : " <<i<<endl; என்பதை இயக்கும். பின்னர், நிரல் கட்டுப்பாட்டு பாய்வு for மடக்கின் மிகுப்பு கோவைக்கு செல்லும். அங்கு i ன் மதிப்பு மிகுக்கப்படும். இதைப்போல 0 முதல் 9 வரை மடக்கு இயக்கப்பட்டு எண்கள் வெளியிடப்படும். நிபந்தனை சோதிப்பு தவறாகும்போது, கட்டுப்பாட்டு பாய்வு மடக்கை விட்டு வெளியேறும்.

எடுத்துக்காட்டு 10.12 for மடக்கை கொண்டு 1 முதல் 10 வரை உள்ள எண்களின் தொடர் கூட்டலை வெளியிடும் நிரல் எழுதுக.

```
#include <iostream>
using namespace std;
int main ()
{
```


Chapter 10.indd 73   03/09/18 4:13 PM

```
int i,sum=0;
for(i=1; i<=10;i++)
 {
          sum=sum+i;
 }
cout<<"The sum of 1 to 10 is "<<sum;
return 0;
}
```

வெளியீடு

```
          The sum of 1 to 10 is 55
```

for மடக்கின் மாறுபட்ட வடிவங்கள்

for கூற்று C++ மொழியில் ஓர் முக்கிய மடக்கு கூற்றாகும். ஏனெனில், இதனை வேறுபட்ட பல வடிவங்களில் பயன்படுத்தலாம். இவ்வகை வடிவங்கள் நெகிழ்வு மற்றும் பொருந்தும் தன்மையை மடக்கிற்கு வழங்குகிறது. for மடக்கின் பல்வேறு வடிவங்கள் கீழே கொடுக்கப்பட்டுள்ளது.

ஒன்றுக்கும் மேற்பட்ட தொடக்க மதிப்பு மற்றும் மிகுப்புகள்:

for மடக்கில் தொடக்க மதிப்பிருத்த மற்றும் மிகுப்பு கோவையை அறிவிக்க ஒன்றுக்கும் மேற்பட்ட கூற்றுகளை பயன்படுத்தலாம். தொடக்க மதிப்பிருத்தல் மற்றும் மிகுப்பு கோவைகளை ஒன்றுக்கும் மேற்பட்ட கூற்றுகளைக் கொண்டு நிறைவேற்றும் போது அவை காற்புள்ளியில் (,) பிரிக்கப்பட்டிருக்க வேண்டும். எடுத்துக்காட்டு:

```
#include<iostream>
using namespace std;
int main()
{
int i j
          for(i=0, j=10 ;  i<j  ;  i++,j--)
{
          cout<<"\nThe value of i is"<<i<<" The value of j is "j;
}
return 0;
}
```

அரைப்புள்ளியால் பிரிக்கப்பட்ட, ஒன்றுக்கும் மேற்பட்ட தொடக்க மதிப்பிருத்தல்கள்.

அரைப்புள்ளியால் பிரிக்கப்பட்ட, ஒன்றுக்கும் மேற்பட்ட மிகுப்பு கோவைகள்.

வெளியீடு

```
          The value of i is 0  The value of j is 10
          The value of i is 1  The value of j is 9
          The value of i is 2  The value of j is 8
          The value of i is 3  The value of j is 7
          The value of i is 4  The value of j is 6
```


Chapter 10.indd 74
03/09/18 4:13 PM

மேலே கொடுக்கப்பட்டுள்ள எடுத்துக்காட்டில் தொடக்க மதிப்பிருத்தலின், i மற்றும் j என்ற இரு வேறு மாறிகள் பயன்படுத்தப்பட்டுள்ளது. அதே போன்று, i++ மற்றும் j++ என்ற இரண்டு மிகுப்புக் கோவைகளை கொண்டுள்ளது. இவ்விரு மாறிகளும் காற்புள்ளியால் பிரிக்கப்பட்டு வரிசை முறையில் வைக்கப்பட்டுள்ளது. மிகுப்பு கோவை i++ முதலில் மிகுக்கப்படும், பின்னர் j++ மிகுக்கப்படும்.

பின்னொட்டு மற்றும் முன்னொட்டு செயற்குறிகள்:

பொதுவாக மிகுப்பு கோவை, மிகுப்பு (++) மற்றும் குறைப்பு (--) செயற்குறியை கொண்டிருக்கும். மிகுப்பு கோவை தனியாக பயன்படுத்தப்படும் போது முன்னொட்டு மிகுப்பு அல்லது குறைப்பு செயற்குறியை விட பின்னொட்டை அதிகம் பொருத்தமானதாக அமையும். முன்னொட்டு செயற்குறி, பின்னொட்டு செயற்குறியை விட விரைவாக இயக்கப்படுவது இம்முறையை விரும்பக் காரணம் ஆகும்.

விருப்ப கோவைகள் (Optional expressions)

பொதுவாக, **for** மடக்கில் மூன்று பகுதிகள் உள்ளது, அவை தொடக்க மதிப்பு, நிபந்தனை கோவைகள் மற்றும் மிகுப்பு கோவை. இந்த மூன்று கோவைகளும், ஓர் மடக்கில் விருப்ப கோவைகளாக பயன்படுத்தப்படலாம்.

சூழல் - 1

எடுத்துக்காட்டு 10.13 (அ) 1 முதல் 10 வரையான எண்களின் கூட்டுத்தொகையை காணும் C++ நிரல்

```
#include <iostream>
using namespace std;
int main ()
{
int i, sum=0, n;
cout<<"\n Enter The value of n";
cin>>n;
i =1;
for (  ; i<=10;i++)
 {
          sum += i;
  }
out<<"\n The sum of 1 to " <<n<<"is "<<sum;
return 0;
}
```

வெளியீடு

```
          Enter the value of n 5
          The sum of 1 to 5 is 15
```

மேலே கொடுக்கப்பட்டுள்ள எடுத்துக்காட்டில், i என்ற கட்டுப்பாட்டு மாறி அறிவிப்பின் போதே அதற்கு 1 என்ற மதிப்பு தொடக்க மதிப்பாக இருத்தப்பட்டுள்ளது. அதாவது, for மடக்கு வெளியே கட்டுப்பாட்டு மாறிக்கு தொடக்க மதிப்பிருத்தப்பட்டுள்ளது. எனினும், for மடக்கினுள் தொடக்க மதிப்பிருத்தும் பகுதியில்

Chapter 10.indd 75 03/09/18 4:13 PM

அரைப்புள்ளி மட்டும் வைக்கப்பட்டு, நிபந்தனை கோவை வரையறுக்கப்பட்டுள்ளது. for மடக்கில் தொடக்க மதிப்பு கொடுக்கப்படாத போது, நிரல் பாய்வு நிபந்தனை சோதிப்புப் பகுதிக்கு அனுப்பி வைக்கப்படும்.

சூழல் - 2

எடுத்துக்காட்டு 10.13 (ஆ) 1 முதல் 10 வரையான எண்களின் கூட்டுத்தொகையை காணும் C++ நிரல்

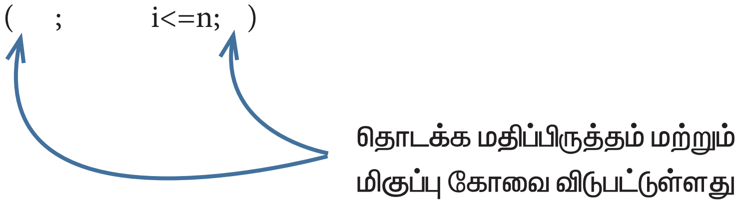
```
#include <iostream>
using namespace std;
int main ()
{
int i, sum=0, n;
cout<<"\n Enter The value of n";
cin>>n;
i =1;
for ( ; i<=10; )
 {
          sum += i;
          ++i;
 }
cout<<"\n The sum of 1 to " <<n<<"is "<<sum;
return 0;
}
```

வெளியீடு

```
          Enter the value of n 5
          The sum of 1 to 5 is 15
```

மேலே கொடுக்கப்பட்டுள்ள நிரலில், தொடக்க மதிப்பிருத்தம் மற்றும் மிகுப்பு கோவை தரப்படவில்லை. ஆனால், இரண்டு பகுதிகளையும் குறிக்க அரைப்புள்ளி மட்டும் பயன்படுத்தப்பட்டுள்ளது.

```
for (    ;         i<=n;   )
```



ஒன்று அல்லது இரண்டு கோவைகள் கொடுக்கப்படாதபோது, நிரல் பாய்வுக் கட்டுப்பாடு சோதிப்பு நிபந்தனைக்கு அனுப்பி வைக்கப்படும்.

சூழல் - 3

for மடக்கில் நிபந்தனை சோதிப்பு கோவை கொடுக்கப்படாத போது மடக்கு முடிவற்றதாக இயக்கப்படும்.

Chapter 10.indd 76 03/09/18 4:13 PM

நிபந்தனை சோதிப்பு விடுபட்டுள்ளது

```
for( i=0 ;   ;     ++i)
    cout<<"\n Welcome";
```

இந்தக் கூற்று முடிவில்லாமல் தொடர்ந்து இயங்கும்.

அதே போன்று, கீழே கொடுக்கப்பட்டுள்ள மடக்கும், முடிவில்லாமல் தொடர்ந்து இயங்கும்.

அனைத்து மூன்று கோவைகளும் விடுபட்டுள்ளது.

```
for(    ;       ;      )
        cout<<"\n Welcome";
```

இந்தக் கூற்று முடிவில்லாமல் தொடர்ந்து இயங்கும்.

வெற்று மடக்கு (Empty loop)

ஒரு மடக்கில் எந்த கூற்றும் இடம்பெறவில்லை என்றால் அது வெற்று மடக்கு எனப்படும்; பின்வரும் மடக்கு ஒரு வெற்று மடக்கு ஆகும்.

```
for( i=0 ; i <=5; +=i) ;
```

for மடக்கின் உடற்பகுதி வெற்று கூற்றுகளை கொண்டுள்ளது.

மேலே கொடுக்கப்பட்டுள்ள மடக்கு வெற்றுக் கூற்றைகொண்டுள்ளதால், இது ஒரு காலி மடக்கு எனப்படும்.

```
int i;
for( i=0       ;       i<=5;   ++i) ;
{
        cout<<"\nWe are Indians";
}
```

for மடக்கின் உடற்பகுதி வெற்று கூற்றுகளைக் கொண்டுள்ளது.

மடக்கு அரைப்புள்ளியில் (;) முடிக்கப்பட்டுள்ளது. எனவே மடக்கின் உடற்பகுதியில் உள்ள கூற்றுகள் எதுவும் இயக்கப்படுவதில்லை.

மேலே கொடுக்கப்பட்டுள்ள நிரலில் for மடக்கின் இறுதியில் அரைப்புள்ளி இடம்பெற்றுள்ளதால், நெளிவு அடைப்புக்குறியில் இடம்பெற்றுள்ள மடக்கின் உடற்பகுதி இயக்கப்படாது.

for மடக்கில் மாறி அறிவிப்பு (Declaration of variable in a for loop)

C++ மொழியில் , மாறிகள் for மடக்குக்கு உள்ளும் அறிவிக்கப்படலாம். எடுத்துக்காட்டு,

Chapter 10.indd 77 03/09/18 4:13 PM

```
int main ()
{
  int sum = 0;
  for(int i=0;  i<=5;  ++i )
  {
        sum = sum + i;
  }
  cout<<"\nThe variable i cannot access here";
  cout<<"\n The variable sum can access here";
}
```

மாறி (i) for மடக்குக்கு உள்ளே அறிவிக்கப்பட்டுள்ளது

மாறி i மடக்கின் உடற்பகுதியில் மட்டுமே செயல்படுத்தப்படும்.

குறிப்பு

main() உள்ளே எதாவது ஒரு தொகுதிக்குள் அறிவிக்கப்படும் மாறிகள், main() தொகுதிக்குள் எங்கு வேண்டுமென்றாலும் அணுகலாம்.

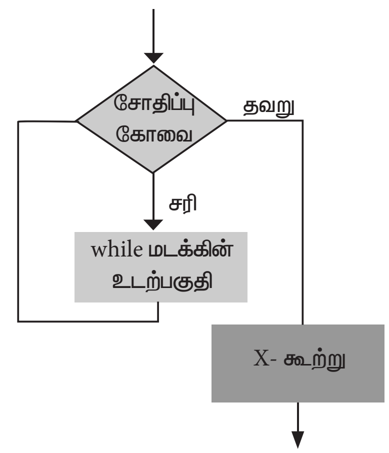
10.5.2 while மடக்கு

while மடக்கும் ஒரு கட்டுப்பாடு பாய்வு கூற்றாகம். இது ஒரு மடக்கினை கொடுக்கப்பட்ட நிபந்தனை சரியாக இருக்கும் வரை, மீண்டும் மீண்டும் இயக்கும். while மடக்கு ஒரு நுழைவு சோதனை மடக்காகும். இதில் சோதிப்பு கோவை முதலில் மதிப்பீடு செய்யப்பட்ட பின்னரே மடக்கினுள் உள்ளே நுழையும்.

while மடக்கின் கட்டளை அமைப்பு :

```
while ( நிபந்தனை சோதிப்புக் கோவை )
{
        மடக்கின் உடற்பகுதி;
}
கூற்று-x;
```

while மடக்கில், நிபந்தனை சோதிப்பு கோவை மதிப்பீடு செய்யப்பட்டு அதன் முடிவு சரி என வந்தால், மடக்கின் உடற்பகுதி இயக்கப்பட்டு மீண்டும் while மடக்கிற்கு கட்டுப்பாடு அனுப்பி வைக்கப்படும்.



பாய்வுப் படம்10.7: while மடக்கின் பாய்வு

நிபந்தனை சோதிப்பு கோவை முடிவு தவறு என வரும்போது, பாய்வுக் கட்டுப்பாடு கூற்று-X அனுப்பி வைக்கப்படும்.

எடுத்துக்காட்டு 10.14 while மடக்கை பயன்படுத்தி 1 முதல் 10 வரை உள்ள எண்களின் தொடர் கூட்டலை வெளியிடும் நிரல் எழுதுக.

```
#include <iostream>
using namespace std;
int main ()
{
int i=1,sum=0;
while(i<=10)
{
        sum=sum+i;
        i++;
}
cout<<"The sum of 1 to 10 is "<<sum;
return 0;
}
```

வெளியீடு

```
The sum of 1 to 10 is 55
```

மேலே கொடுக்கப்பட்டுள்ள நிரலில், முழு எண் மாறி i யின் மதிப்பு 1 என்றும் sum என்ற மாறியின் மதிப்பு 0 என்றும் மதிப்பிருத்தப்பட்டுள்ளது. while மடக்கு i < 10 என்ற நிபந்தனையை சோதித்து, அதன் முடிவு i என்ற மாறியின் மதிப்பு sum மாறியுடன் கூட்டப்படும், பின்னர் i யின் மதிப்பு 1 மிகுக்கப்படும். மீண்டும் நிபந்தனை i < 10 சோதிக்கப்படும். இப்பொழுது 2 < 10, சரி என்பதால் sum என்ற மாறியியுடன் 2 கூட்டப்படும். இது போன்று i யின் மதிப்பு 11 ஆகும் வரை நடைபெறும். இப்பொழுது 11 < 10 என்ற சோதனை தவறு என வருவதால், while மடக்கை விட்டு வெளியே செல்லும். மடக்கை விட்டு வெளியே வந்ததும், sum மாறியின் மதிப்பு வெளியிடப்படும்.

எடுத்துக்காட்டு 10.15 while மடக்கை பயன்படுத்தி 5 எண்களை உள்ளீடாகப் பெற்று அவற்றின் கூட்டுத் தொகை மற்றும் சராசரியை காணும் நிரல் எழுதுக.

```
#include <iostream>
using namespace std;
int main ()
{
int i=1, num, avg, sum=0;
while (i <= 5)
{
        cout<<"Enter the number : ";
        cin>>num;
        sum=sum+num;
        i++;
}
avg=sum/50;
cout<<"The sum is "<<sum<<endl;
```


Chapter 10.indd 79 03/09/18 4:13 PM

```
cout<<"The average is "<<avg;
return 0;
}
```

வெளியீடு

```
Enter the number : 1
Enter the number : 2
Enter the number : 3
Enter the number : 4
Enter the number : 5
The sum is 15
The average is 3
```

மேலே கொடுக்கப்பட்டுள்ள நிரலில், num , avg என்ற மாறி அறிவிக்கப்பட்டு அதன் மதிப்பு 1 மற்றும் sum மதிப்பு 0 யின் தொடங்கி வைக்கப்பட்டுள்ளது. while மடக்கு, i <= 5 என்ற நிபந்தனையை சோதித்து, சரி என முடிவு பெறப்பட்டால், பயனரிடம் இருந்து ஒரு எண் உள்ளீடாகப் பெறப்பட்டு, i மதிப்பு மிகுக்கப்படும். மேலும் i <= 5 என்ற நிபந்தனையை சோதித்து, சரி என முடிவு பெறப்பட்டால், பயனரிடம் இருந்து ஒரு எண் உள்ளீடாகப் பெறப்பட்டு, i யின் மதிப்பு மிகுக்கப்படும். மேலும் 2 <=5 என்ற சோதிப்பு போது இரண்டாவது எண் பயனரிடம் பெறப்பட்டு sum உடன் கூட்டப்படும். இது போன்று I யின் மதிப்பு 6 வரும் வரை இயக்கப்படும். இந்நிலையில் while மடக்கு மடக்கை விட்டு வெளியேறுவதும் மடக்கை விட்டு வெளியேறுவதும் avg மற்றும் sum கணக்கிடப்பட்டு வெளியிடப்படும்.

while மடக்கின் மாறுபட்ட வடிவங்கள்

while மடக்கு பல்வேறு வடிவங்கள் பெற்றிருக்கலாம். அது வெறுமை மடக்காகவோ அல்லது முடிவற்ற மடக்காகவோ இருக்கலாம். ஒரு வெறுமை while மடக்கின் உடற்பகுதியில் null கூற்றைத் தவற வேறு எந்த கூற்றும் இடம் பெற்றிருக்காது. Null கூற்று என்பது அரைப்புள்ளி (;) ஆகும். எடுத்துக்காட்டு

```
int main()
{

    int i=0;
    while(++i < 10000 )

    return 0;

}
```

→ இந்த மடக்கு எந்த கூற்றையும் பெற்றிருக்கவில்லை. எனவே இது ஒரு காலி மடக்காகும்.

மேலே கொடுக்கப்பட்டுள்ள குறிமுறையில், மடக்கு ஒரே தாமத மடக்காக (delay loop) கொடுக்கப்பட்டுள்ளது. நேர தாமத மடக்கு ஒரு நிரலை சில நோடிகள் இடைநிறுத்தப் பயன்படுகிறது.

எடுத்துக்காட்டு

Chapter 10.indd 80 03/09/18 4:13 PM

```
int main()
{
    int i = 0;
    while(i < =10)
      cout <<"The value of i is "<<i;
       i++;
     return 0;
}
```

இந்த கூற்று கொடுக்கப்பட்ட மதிப்பை முடிவில்லாமல் தோன்ற செய்யும் ஏனெனில் மடக்கின் உடற்பகுதியில் மிகுப்பு கூற்றில் இடம் பெற்றிருக்கவில்லை

இந்த கூற்று while மடக்கின் பகுதியாக கருதப்படுவதில்லை ஏனெனில் இது நெளிவு அடைப்புக் குறிக்குள் கொடுக்கப்படவில்லை

இதே போன்று, while மடக்கை வேறொரு மாறுபட்ட வடிவம் கீழே கொடுக்கப்பட்டுள்ளது

```
int main()
{
    int i=1;
    while( ++i  < 10)
    cout<< "The value of i is "<<i;
    return 0;
}
```

மேற்கண்ட நிரலில், while (++i < 10), என்ற கூற்றில் முதலில் i என்ற மாறியின் மதிப்பு மிகுக்கப்படுகிறது. பின்னர், i மாறியின் மதிப்பு 10 வுடன் ஒப்பீடு செய்யப்பட்டு, cout << The value of i is " << i; என்ற கூற்றை நிபந்தனைக் கட்டுப்பாடு நிறைவடையும் வரை i என்ற மாறியின் மதிப்பு திரையிடப்படும்.

```
int main()
{
    int i=1;
    while( i++  < 10)
    cout<< "The value of i is "<<i;
    return 0;
}
```

மேற்கண்ட நிரலில், while (i ++ < 10), என்ற கூற்றில் முதலில் i, என்ற மாறியின் மதிப்பு 10 யுடன் ஒப்பீடு செய்யப்பட்ட பின், i என்ற மாறியின் மதிப்பு மிகுக்கப்படுகிறது. cout << The value of i is " << i; என்ற கூற்றை கட்டுப்பாடு அடையும் போது i என்ற மாறியின் மதிப்பு செய்யப்பட்டிருக்கும்.

10.5.3 do-while மடக்கு

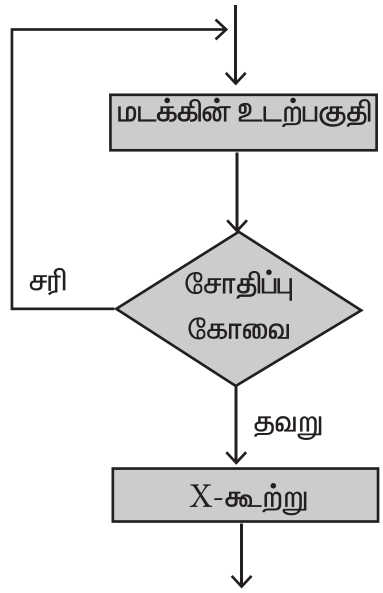
do-while மடக்கு வெளியேறல் சோதிப்பு மடக்காகும். do while மடக்கினில் மடக்கின் உடற்பகுதி செயல்படுத்தப்பட்ட பின் மடக்கின் இறுதியில் நிபந்தனை சரிபார்க்கப்படும். ஆகையால், நிபந்தனை தவறு என மதிப்பிடப்படும் போதும் மடக்கின் உடற்பகுதி ஒரு முறையேனும் நிறைவேற்றப்படும்.

do-while மடக்கின் கட்டளை அமைப்பு:

```
do
{
        மடக்கின் உடற்பகுதி;
} while(நிபந்தனை சோதிப்பு கோவை);
```

do-while மடக்கின் பாய்வுக் கட்டுப்பாடு மற்றும் பாய்வு படம் கீழே கொடுக்கப்பட்டுள்ளது.

Chapter 10.indd 81 03/09/18 4:13 PM



பாய்வுப் படம் 10.13: do-while மடக்கின் பாய்வு கட்டுப்பாடு

எடுத்துக்காட்டு 10.16 do-while() மடக்கினைப் பயன்படுத்தி 10 முதல் 1 வரை உள்ள எண்களை வெளியிட நிரல் எழுதுக.

```
#include <iostream>
using namespace std;
int main ()
{
int n = 10;
do
{
        cout<<n<<", ";
        n--;
}while (n>0) ;
}
```

வெளியீடு

```
        10, 9, 8, 7, 6, 5, 4, 3, 2, 1,
```

மேற்கண்ட நிரலில், n என்ற முழு எண் மாறியின் தொடக்க மதிப்பு 10 என இருத்தப்படுகிறது. அடுத்து, n ன் மதிப்பு 10 என வெளியிடப்பட்டு, பின் 1 குறைக்கப்படும். இப்போது 9 > 0 என்ற நிபந்தனை சரிபார்க்கப்பட்டு ,9 என்று வெளியிடப்படும். பின் n என்பது 8 ஆக குறைக்கப்படும். நன் மதிப்பு 0 என ஆகும் வரை இந்த செயல்பாடு தொடரும். நிபந்தனை n > 0 என்பது தவறு என ஆகும் போது, do-while மடக்கு முடிவு பெறும்.

10.5.4 பின்னலான மடக்குகள் (Nesting of loops)

ஒரு மடக்கினுள் இன்னொரு மடக்கு இடம்பெறுமெனில் அது பின்னலான மடக்கு எனப்படும்.

கட்டளை அமைப்பு கீழே தரப்பட்டுள்ளது :‘

```
for (தொடக்க மதிப்பு; நிபந்தனைக் கோவை; மிகுப்பு கோவை)
{
        for (தொடக்க மதிப்புதொடக்க மதிப்பு;நிபந்தனைக் கோவை; மிகுப்பு கோவை)
          {
          கூற்று / கூற்றுகள்;
           }
கூற்று / கூற்றுகள்;
}
```

```
while(நிபந்தனை)
{
        while(நிபந்தனை)
        {
        கூற்று / கூற்றுகள்;
        }
கூற்று / கூற்றுகள்;
}
```

```
do
{
கூற்று / கூற்றுகள்;
        do
        {
        கூற்று / கூற்றுகள்;
        }while(நிபந்தனை);
} while(நிபந்தனை);
```

எடுத்துக்காட்டு 10.17 பின்னலான for மடக்கினை கொண்டு அணிக்கோவையில் பெருக்கல் வாய்ப்பாட்டினை வெளியிட ஒரு நிரலை எழுது.

```
#include<iostream>
using namespace std;
int main(void)
{
        cout<< "A multiplication table:" <<endl
        <<"  1\t2\t3\t4\t5\t6\t7\t8\t9" <<endl<< "" <<endl;
        for(int c = 1; c < 10; c++)
        {
        cout<< c << "| ";
        for(int i = 1; i< 10; i++)
                {
        cout<<i * c << '\t';
        }
        cout<<endl;
        }
return 0;
}
```

வெளியீடு

A multiplication table:

	1	2	3	4	5	6	7	8	9
1 |	1	2	3	4	5	6	7	8	9
2 |	2	4	6	8	10	12	14	16	18
3 |	3	6	9	12	15	18	21	24	27
4 |	4	8	12	16	20	24	28	32	36
5 |	5	10	15	20	25	30	35	40	45
6 |	6	12	18	24	30	36	42	48	54
7 |	7	14	21	28	35	42	49	56	63
8 |	8	16	24	32	40	48	56	64	72
9 |	9	18	27	36	45	54	63	72	81

10.6 தாவுதல் கூற்றுகள் (Jump statements)

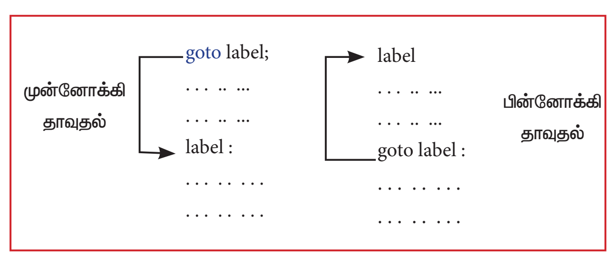
தாவுதல் கூற்று நிரலின் ஓட்டத்தை குறுக்கிட உதவுகிறது. தாவுதல் கூற்றின் வகைகள்

- goto கூற்று
- break கூற்று
- continue கூற்று

10.6.1 goto கூற்று

goto கூற்று நிபந்தனையில்லா கட்டுப்பாட்டு கூற்றாகும். இது நிரலில் ஒரு கூற்றிலிருந்து மற்றொரு கூற்றிற்கு எந்த வித நிபந்தனையும் இன்றி தாவச் செய்யும்.

கட்டளை அமைப்பு 1	கட்டளை அமைப்பு 2
goto label;	**label:**
-------------	------------
-------------	------------
-------------	------------
label:	**goto label;**



மேலே கொடுக்கப்பட்டுள்ள கட்டளை அமைப்பில், முகவரி என்பது குறிப்பெயராகும்- goto label; என்ற கூற்று இயக்கப்படும் போத கட்டுப்பாடு label: என்ற கூற்றினுக்கு தாவும் பின் அதற்கு கீழ் வரும் இடம் பெற்றுள்ள கூற்றுகள் இயக்கப்படும்.

Chapter 10.indd 84 03/09/18 4:13 PM

எடுத்துக்காட்டு : 10.18 goto கூற்றை பயன்படுத்தி கொடுக்கப்பட்ட எண்களின் சராசரியை கணக்கிடும் C++ நிரல் எழுதுக.

```
# include <iostream>
using namespace std;
int main()
{
float num, average, sum = 0.0;
int i, n;
cout<< "Maximum number of inputs: ";
cin>> n;
for(i = 1; i<= n; ++i)
{
cout<< "Enter n" <<i<< ": ";
cin>>num;
if(num< 0.0)
{
// Control of the program move to jump:
        goto jump;
}
sum += num;
}
jump:
average = sum / (i - 1);
cout<< "\nAverage = " << average;
return 0;
}
```

வெளியீடு

```
        Maximum number of inputs: 5
        Enter n1: 10
        Enter n2: 20
        Enter n3: -2
        Average = 15
```

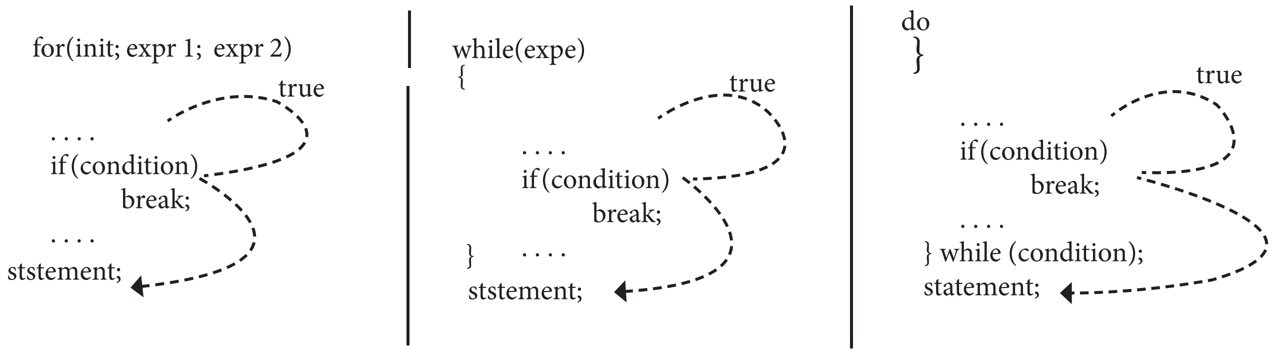
மேலே கொடுக்கப்பட்டுள்ள நிரலில் பயனர் உள்ளிடும் எண்களின் சராசரி கணக்கிடப்படுகிறது. பயனர் எதிர்மறை எண்ணை உள்ளிடும் போது, எண்களின் சராசரி கணக்கீடு இயக்கப்படாது.

10.6.2 break கூற்று

break என்ற தாவுதல் கூற்று கட்டுப்பாட்டை மடக்கினை விட்டு வெளியேற செய்யும், மேலும் மடக்கினுள் உடற்பகுதிக்கு அடுத்துள்ள கூற்றுகளை இயங்கச் செய்யும்.

கீழே கொடுக்கப்பட்டுள்ள படம் ஓர் மடக்கில் break கூற்று இயங்கும் முறையை காட்டுகிறது.

Chapter 10.indd 85 03/09/18 4:13 PM



படம் 10.14: break கூற்று for, while மற்றும் do-while loop

எடுத்துக்காட்டு 10.19 break கூற்றை பயன்படுத்தி எண்ணிக்கையை காணும் நிரல்

```
#include <iostream>
Using namespace std;
int main ()
{
int count = 0;
do
{
cout<< "Count : " << count <<endl;
count++;
if( count > 5)
 {
        break;
 }
}while( count < 20 );
return 0;
}
```

வெளியீடு

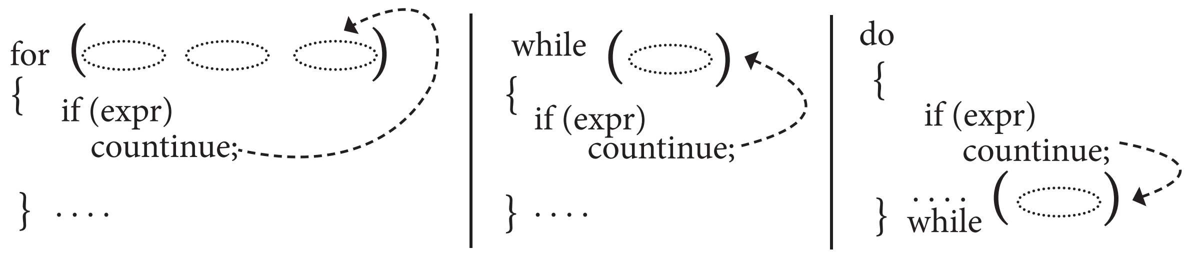
```
        Count : 0
        Count : 1
        Count : 2
        Count : 3
        Count : 4
        Count : 5
```

மேலே கொடுக்கப்பட்டுள்ள எடுத்துக்காட்டில், while மடக்கு 20 முறை இயக்கப்படும் ஆனால் countன் மதிப்பு ஆகிய பின் break கூற்றால் மடக்கை முடித்து வைக்கும்.

Chapter 10.indd 86 03/09/18 4:13 PM

10.6.3 continue கூற்று

Continue கூற்று break கூற்றைப்போல் செயல்படுகிறது. ஆனால், மடக்கை முடித்து வைப்பதற்கு பதிலாக, மடக்கை அடுத்து சுழற்சிக்கு இட்டுச்செல்லும். ஓர் மடக்கினுள் Continue கூற்று இயக்கப்படும் பொழுது, continue கூற்றை தொடர்ந்து வரும் கூற்றுகள் தவிர்க்கப்பட்டு, மடக்கின் அடுத்த சுழற்சியை தொடங்க செய்யும்.



continue கூற்றின் பாய்வு

எடுத்துக்காட்டு 10.20 continue கூற்றை பயன்படுத்தி ஒன்று முதல் பத்து வரையான எண்களை அச்சிடும் C++ நிரல் எழுதுக.

```
#include <iostream>
using namespace std;
 int main()
{
for (int i = 1; i<= 10; i++)
 {
if (i == 6)
continue;
else
cout<<i<< " ";
}
return 0;
}
```

வெளியீடு

```
1 2 3 4 5 7 8 9 10
```

மேலே கொடுக்கப்பட்டுள்ள எடுத்துக்காட்டில் for மடக்கு 0 முறை இயக்கப்படும், ஆனால் i மாறி 6 ஆகும் போது continue கூற்றால் கட்டுப்பாடு for மடக்கிற்கு அனுப்பி வைக்கப்படும்.

break மற்றும் continue ஒப்பீடு

Break	Continue
break மடக்கினை விட்டு வெளியேற செய்யும்.	continue கூற்று மடக்கை விட்டு வெளியேற செய்யாது.
இது பன்முறை செயலை முடித்து வைக்கும்.	இது பன்முறை செயலை தவிர்க்கச் செய்யும்.

Chapter 10.indd 87 03/09/18 4:13 PM

இந்த கூற்றை இயக்கப்படும் பொழுது கட்டுப்பாட்டை மடக்கினை விட்டு வெளியேறி மடக்கிற்கு அடுத்து வரும் கூற்றுகளை இயக்கும்.	இந்த கூற்று இயக்கப்படும் பொழுது மடக்கைவிட்டு வெளியேறாமல் மடக்கின் அடுத்த சுழற்சிக்கு தாவச் செய்யும்.
break கூற்று மடக்கு மற்றும் switch கூற்றுடன் பயன்படுத்தலாம்.	continue கூற்று மடக்கில் மட்டுமே பயன்படுத்தப்படும், switch கூற்றுடன் பயன்படுத்தப்படாது.

செய்முறைப் பயிற்சி

கீழ்காணும் சிக்கல்களை தீர்க்கும் C++ நிரல்களை எழுதுக.

1. வெப்பநிலை அளவு மாற்ற நிரல்: கொடுக்கப்படும் வெப்பநிலை அளவை, பயனரின் விருப்பத்திற்கு ஏற்ப ஃபாரன்ஹீட்டிலிருந்து செல்ஷியஸ்-க்கும்; செல்ஷியஸிலிருந்து ஃபாரன்ஹீட்டுக்கும் மாற்றம் செய்யும் C++ நிரல் ஒன்றை எழுதுக.

2. பயனரிடம், இரண்டு எண்களையும், ஒரு செயற்குறியையும் உள்ளீடாகப் பெற வேண்டும். பெறப்பட்ட செயற்குறியை அடிப்படையாகக் கொண்டு, கூட்டல், கழித்தல், பெருக்கல் அல்லது வகுத்தல் கணக்கீட்டை கொடுக்கப்பட்ட இரண்டு எண்களின் மீது செய்து வெளியீட்டை காட்டும் C++ நிரல் ஒன்றை எழுதுக.

3. C++ குறியுருத் தொகுதியிலுள்ள ஏதேனும் ஒரு குறியுருவை உள்ளீடாகப் பெற வேண்டும். பெறப்பட்ட குறியுரு எழுத்து, எண் அல்லது சிறப்புக் குறியுரு இவற்றில் எது என்பதைக் காட்டும் C++ நிரல் ஒன்றை எழுதுக.

4. கொடுக்கப்படும் குறியுரு ஆங்கில பெரிய எழுத்தா, சிறிய எழுத்தா, எண்ணா அல்லது வேறு ஏதேனும் குறியுருவா என்பதை ASCII மதிப்புகளைக் கொண்டு கண்டறியும் C++ நிரல் ஒன்றை எழுதுக. ASCII மதிப்புகள் கீழே கொடுக்கப்பட்டுள்ளது.

குறியுருக்கள்	**ASCII** மதிப்புகள்
'0' - '9'	48 - 57
'A' - 'Z'	65 - 90
'a' - 'z'	97 - 122
மற்ற குறியுருக்கள்	0- 255 மேலே குறிப்பிடப்பட்டுள்ள மதிப்புகள் நீங்கலாக

5. ஒரு முழு எண்ணின் காரணிப்படுத்தலை (Factorial) கணக்கிடும் C++ நிரல் ஒன்றை எழுதுக.

6. 1 2 4 8 16 32 64 128 என்ற எண் தொடரை அச்சிடும் C++ நிரல் ஒன்றை எழுதுக.

7. கொடுக்கப்படும் ஒரு எண்ணின் வகுத்திகளை கண்டறியும் C++ நிரல் ஒன்றை எழுதுக.

8. ஃபிபோனாசி எண் வரிசை (0 1 1 2 3 5 8 …….) கண்டறியும் C++ நிரல் ஒன்றை எழுதுக.

Chapter 10.indd 88 03/09/18 4:13 PM

9. பின்னலான மடக்குகளைப் பயன்படுத்தி, பின்வரும் வடிவங்களை அச்சிடுவதற்கான C++ நிரல் ஒன்றை எழுதுக.

(அ)	(ஆ)	(இ)
A	5 4 3 2 1	# # # # # # #
A B	5 4 3 2	# # # # #
A B C	5 4 3	# # #
A B C D	5 4	#
A B C D E	5	
A B C D E F		

10. கொடுக்கப்பட்ட ஒரு எண்ணின் வர்க்க மூலம், பகு எண்ணா? இல்லையா எனக்காணும் C++ நிரல் ஒன்றை எழுதுக.

நினைவில் கொள்க

- கணிப்பொறி நிரல் என்பது ஒரு குறிப்பிட்ட பணியை நிறைவேற்ற கொடுக்கப்படும் கூற்றுகள் அல்லது கட்டளைகளின் தொகுப்பாகும்.
- C++ நிரலில் இரு வகையான கூற்றுகள் உள்ளன. (1) வெற்றிட கூற்றுகள் (2) கலவை கூற்றுகள்
- கட்டுப்பாடு, நிரலின் ஒரு பகுதியிலிருந்து இன்னொரு பகுதிக்குத் தாவுவதற்குக் காரணமாக நிரல் கூற்றுகள் கட்டுப்பாட்டு கட்டமைப்புகள் எனப்படுகின்றன.
- C++ ல் மூன்று வகையான கட்டளை கூற்றுகள் உள்ளன: (1) வரிசை முறை கூற்றுகள் (Sequence Statement) (2) தேர்ந்தெடுப்பு கூற்றுகள் (Selection Statement) (3) சுழற்சி முறை (Iteration Statement)
- if மற்றும் switch கூற்றுகள் தேர்வு கூற்றுகள் ஆகும்.
- இரு மாற்று வழிகளுள், ஒன்றைத் தேர்ந்தெடுக்கும் if .. else... கட்டளை அமைப்பில் நிபந்தனை அடிப்படையில் ஒரு குறிப்பிட்ட கட்டளைத் தொகுதி மட்டுமே நிறைவேற்றப்படும்.
- Switch கூற்று ஒரு பல வழி கிளைப் பிரிப்பு கூற்றாகும்.
- மடக்குகள், ஒரு கட்டளைத் தொகுதியை, நிபந்தனையின் அடிப்படையில் குறிப்பிட்ட தடவைகள் திரும்ப திரும்ப நிறைவேற்றுகின்றன.
- C++ மொழியில் மூன்று வகையான பன்முறைச் செயல் அல்லது மடக்குகள் உள்ளன. For, while,do-while
- C++ தாவும் (Jump) கூற்றுகள் Goto, Break,continue ஆகும்.

மதிப்பீடு

N18DAT

பகுதி-அ

சரியான விடையை தேர்வு செய்யவும் :

1. வெற்றுக்கூற்றின் மாற்றுப் பெயர் என்ன?

   (அ) கூற்று அல்லா (ஆ) காலிக் கூற்று

   (இ) void கூற்று (ஈ) சுழியக் கூற்று

2. C++ல் குறிமுறைத் தொகுதிகள் இந்தக் நிறுத்தற்குறிக்குள் கொடுக்கப்பட வேண்டும்:

   (அ) { } (ஆ) [ ] (இ) ( ) (ஈ) < >

3. சுழற்சியில், மீண்டும் மீண்டும் இயக்கப்படும் குறிமுறைத் தொகுதிகள் இவ்வாறு அழைக்கப்படுகிறது:

   (அ) நிபந்தனை (ஆ) மடக்கு (இ) கூற்று (ஈ) மடக்கின் உடற்பகுதி

4. பல வழி கிளைப் பிரிப்புக் கூற்று:

   (அ) if (ஆ) if … else (இ) switch (ஈ) for

5. சுழற்சிக் கூற்றுகள் எத்தனை வகைப்படும்?

   (அ) 2 (ஆ) 3 (இ) 4 (ஈ) 5

6. for (int i=0; i<10; i++) என்ற மடக்கு எத்தனை முறை இயங்கும்?

   (அ) 0 (ஆ) 10 (இ) 9 (ஈ) 11

7. பின்வருவனவற்றில் எது வெளியேறல் சோதிப்பு மடக்கு?

   (அ) for (ஆ) while (இ) do…while (ஈ) if…else

8. தாவுதல் கூற்றுகளின் சிறப்புச் சொற்களில் பொருந்தா ஒன்றை கண்டுபிடி.

   (அ) break (ஆ) switch (இ) goto (ஈ) continue

9. பின்வருவனவற்றில் எது நுழைவு சோதிப்பு மடக்கு?

   (அ) for (ஆ) while (இ) do…while (ஈ) if…else

10. ஒரு மடக்கு அதன் உடற்பகுதியில் மற்றொரு மடக்கை பெற்றிருப்பது:

    (ஆ) பின்னலான மடக்கு (ஆ) உள் மடக்கு

    (இ) உள்ளிணைந்த மடக்கு (ஈ) மடக்குகளின் பின்னல்

Chapter 10.indd 90 03/09/18 4:13 PM

பகுதி - ஆ

குறு வினாக்கள் (2 மதிப்பெண்கள்) :

1. வெற்றுக்கூற்று மற்றும் கூட்டுக்கூற்று என்றால் என்ன?
2. தேர்ந்தெடுப்புக் கூற்றுகள் என்றால் என்ன? அதன் வகைகளை எழுதுக.
3. பின்வரும் நிரலில் உள்ள பிழைகளை சரிசெய்க:

```
if (x=1)
  p= 100;
else
  p = 10;
```

4. பின்வரும் நிரலின் வெளியீடு என்ன?

```
int year;
cin >> year;
  if (year % 100 == 0)
          if ( year % 400 == 0)
                  cout << "Leap";
  else
          cout << "Not Leap year";
```

If the input given is (i) 2000 (ii) 2003 (iii) 2010?

5. பின்வரும் நிரலில் உள்ள பிழைகளை சரிசெய்க:

```
for (int i=2; i<=10 ; i+=2)
  cout << i;
```

6. 21 முதல் 30 வரை தொடர்ச்சியாக எண்களை அச்சிடுவதற்கான for மடக்கை எழுதுக.
7. 2, 4, 6, 8 .... 20 என்ற தொடர் வரிசையை அச்சிடுவதற்கான while மடக்கை எழுதுக.
8. if கூற்றுடன், ?: மும்ம செயற்குறியை ஒப்பிடுக.

பகுதி- இ

சிறு வினாக்கள் (3 மதிப்பெண்கள்) :

1. பின்வரும் if - else கூற்றுக்கு நிகரான நிபந்தனை கூற்றாக மாற்றுக:

```
if (x >= 10)
        a = m + 5;
else
        a = m;
```

2. பின்வரும் நிரல் கூற்றுகள் சரியாக இயங்கும் வகையில் அவற்றை மாற்றி எழுதுக.

```
v = 5;
do;
{
  total += v;
  cout << total;
while v <= 10
```

3. கொடுக்கப்பட்ட எண்ணின் பெருக்கல் வாய்ப்பாட்டை அச்சிடும் C++ நிரல் ஒன்றை எழுதுக.

4. switch கூற்றின் கட்டளை தொடரை எழுதி அதன் பயன்களை பட்டியலிடுக.

5. பின்வரும் எண் தொடரை அச்சிடுவதற்கான நிரல்களை எழுதுக.

   (a) 1 4 7 10...... 40

பகுதி – ஈ

பெரு வினாக்கள் (5 மதிப்பெண்கள்) :

1. கட்டுப்பாட்டு கூற்றுகளை பொருத்தமான எடுத்துக்காட்டுடன் விளக்குக.

2. நுழைவு சோதிப்பு மடக்கு என்றால் என்ன? ஏதேனும் ஒரு நுழைவு சோதிப்பு மடக்கை பொருத்தமான எடுத்துக்காட்டுடன் விளக்குக.

3. LCM மற்றும் GDC போன்றவற்றை கணக்கிடுவதற்கான நிரல்களை எழுதுக.

4. கீழ்காணும் எண் தொடர்களை கணக்கிடுவதற்கான நிரல்களை எழுதுக

   (a) $x - \frac{x^2}{2!} + \frac{x^3}{3!} - \frac{x^4}{4!} + \frac{x^5}{5!} - \frac{x^6}{6!}$

   (b) $x + \frac{x^2}{2} + \frac{x^3}{3} + ... + \frac{x^n}{n}$

5. கொடுக்கப்பட்ட எண் தொடரின் கூட்டுத் தொகையை கணக்கிடும் நிரல் ஒன்றை எழுதுக.

   $S = 1 + x + x^2 + ... + x^n$

துணை நூல்கள்:

(1) Object Oriented Programming with C++ (4th Edition), Dr. E. Balagurusamy, Mc.Graw Hills.

(2) The Complete Reference C++ (Forth Edition), Herbert Schildt. Mc.Graw Hills.

(3) Computer Science with C++ (A text book of CBSE XI and XII), Sumita Arora, Dhanpat Rai & Co.

பாடம் 11

C++ - ன் செயற்கூறுகள்

கற்றலின் நோக்கங்கள் :

இந்தப் பாடப்குதியைக் கற்றபின் மாணவர் அறிந்து கொள்வது.

- செயற்கூறுகளின் வரையறை மற்றும் செயற்கூறுகளின் பயன்களைப் பற்றி அறிந்து கொள்ளுதல்.
- முன்னரே வரையறுக்கப்பட்ட மற்றும் பயனர் வரையறுக்கும் செயற்கூறுகளை பற்றி அறிந்து கொள்ளுதல்.
- சிக்கல்களைத் தீர்க்க கணித செயற்கூறுகளைப் பயன்படுத்துதல்.
- சரம் மற்றும் குறியுறு செயற்கூறுகளைப் பயன்படுத்தி சரம் மற்றும் குறியுறு தரவுகளை கையாளுதல் .
- பெரியசிக்கல்களை சிறுசிறுசெயற்கூறுகளாக உருவாக்கி அவற்றை செயல்படுத்தும் நிரலாக்கம் (**Modular Programming**) பற்றி அறிந்து கொள்ளுதல்.
- செயலுருபுகளின் பங்குகள் மற்றும் செயலுருபுகளின் பல்வேறு வழிமுறைகளை ஒப்பிடுதல்.
- மாறிகள் மற்றும் செயற்கூறுகளின் வரையெல்லையைக் கண்டறிதல்.

11.1 முன்னுரை

ஒரு பெரிய நிரலை சிறிய துணை நிரலாக பிரிக்க முடியும். அவ்வாறு பிரிக்கப்படும் அத்தணை நிரல் செயற்கூறு என்று அழைக்கப்படுகிறது. ஒவ்வொரு செயற்கூறும் அதற்குரிய செயற்பாடுகளை செயல்படுத்தும். நிரலின் நீளத்தையும் மற்றும் சிக்கற்பாட்டையும் குறைக்கவும் நிரலை எளிதில் புரிந்து கொள்ளவும், பிழைகளைக் கண்டறிந்து திருத்தவும் செயற்கூறுகள் வழிவகுக்கிறது. தானமைவு செயற்கூறுகள் "உள்ளமைந்த செயற்கூறுகள் (built – in functions)" என்றும் பயனர் உருவாக்கிய செயற்கூறுகள் "பயனர் வரையறுத்த செயற்கூறுகள்" என்றும் அழைக்கப்படுகிறது.

- உள்ளமைந்த செயற்கூறுகள் – C++ மொழியின் பொதுக்களஞ்சியத்தில் (**Standard Library**) உள்ள செயற்கூறுகளாகும்.
- பயனர் வரையறுத்த செயற்கூறுகள் – பயனர் உருவாக்கிய செயற்கூறுகளாகும்.

11.2 செயற்கூறுகளின் தேவை

நிரலின் நீளத்தை மற்றும் சிக்கற்பாட்டை குறைப்பதற்கு செயற்கூறுகள் பயன்படுத்தப்படுகின்றது. நிரலர்கள் அவர்களுடைய சொந்த செயற்கூறுகளை எழுதியோ அல்லது நிலையான நூலகத்திலுள்ள செயற்கூறுகளைப் பயன்படுத்தியோ துணை நிரல்களை (Sub Programs) உருவாக்கலாம்.

1. பிரித்தலும் மற்றும் சேர்த்தலும் (Divide and Conquer)

- சிக்கலான நிரல்களை மேலாண்மை செய்ய அவற்றை துணை நிரல்களாகப் பிரிக்கலாம்.
- தனிப்பட்ட செயற்கூறுகளை உருவாக்குதல், பிழை கண்டறிதல் மற்றும் சோதனை செய்தல் போன்றவற்றில் நிரலர்கள் கவனம் செலுத்தலாம்.
- பல நிரலர்கள் ஒரே நேரத்தில் பல்வேறு செயற்கூறுகளில் பணியாற்றலாம்.

Chapter 11.indd 93 03/09/18 5:47 PM

2. மறுபயனாக்கம் (Reusability):

- நிரலில் உள்ள சில வரிகளை மீண்டும் மீண்டும் வெவ்வேறு சந்தர்ப்பங்களில் பயன்படுத்தலாம். ஒரு நிரலின் பல்வேறு நகல்களை நீக்க செயற்கூறுகள் பயன்படுகின்றன. மேலும் செயல்கூறுகள் நிரலை பராமரிக்கவும் அவற்றின் அளவை குறைக்கவும் பயன்படுகின்றது. சில செயற்கூறுகளை எத்தனை முறை வேண்டுமானாலும் பல்வேறு உள்ளீடுகளை கொண்டு அழைக்கலாம்.

11.3 செயற்கூறுகளின் வகைகள்

செயற்கூறுகளை இரு வகைகளாக பிரிக்கலாம்.

1. முன் வரையறுக்கப்பட்ட (அல்லது) உள்ளமைந்த (அல்லது) நூலக செயற்கூறுகள்.

2. பயனர் வரையறுக்கப்பட்ட செயற்கூறுகள்.

பல்வேறு செயற்பாட்டிற்கு உடனே பயன்படுத்தும் வகையில் C++ மொழியில் உயரிய சேகரிப்புகளாக பல செயற்கூறுகள் உள்ளன. தலைப்பு கோப்புகளில் இவ்வகை செயற்கூறுகளின் வரையறைகளை முன்னரே எழுதப்பட்டு, பிழை திருத்தி மற்றும் நிரல் பெயர்க்கப்பட்ட (Complied) அவற்றைத் தொகுத்து சேமிக்கப்பட்டுள்ளன. இவ்வாறு நம் தேவைக்கு உடனே உபயோகிக்கப்படுத்தப்படும் துணை நிரல்களை முன் வரையறுக்கப்பட்ட செயற்கூறுகள் அல்லது உள்ளமைந்த செயற்கூறுகள் என்றழைக்கப்படுகின்றன. குறிப்பிட்ட பணிக்கான புதிய செயற்கூறுகளை பயனர் தேவைகேற்ப உருவாக்கும் வசதிகள் C++ மொழியில் உள்ளது. அவ்வாறு உருவாக்கும் பணிக்கான பெயர் மற்றும் தரவுகளின் தேவை (செயலுருப்புகள்) போன்றவற்றை பயனரே தீர்மானிப்பதால் இவ்வகை செயற்கூறுகளை பயனர் வரையறுத்த செயற்கூறுகள் என்கின்றோம்.

11.4 C++ -ல் உள்ள தலைப்பு கோப்புகள் மற்றும் உள்ளமைந்த செயற்கூறுகள்

நூலக செயற்கூறுகளுக்கு தேவையான செயற்கூறு முன்மாதிரி மற்றும் வரையறுப்புகளை தலைப்புக் கோப்புகள் கொண்டுள்ளது. நூலக செயற்கூறுகளில் உபயோகிக்கக்கூடிய தரவின் வகைகள் மற்றும் மாறிலிகளை பற்றிய வரையறுப்புக்களும் தலைப்புக்கோப்பு கொண்டுள்ளது. தலைப்பு கோப்பின் விரிவாக்கம். h என்று அறியப்படும். ஒரு தலைப்புக்கோப்பில் பல உள்ளிணைந்த செயற்கூறுகளைப் பற்றிய வரையறுப்புகள் கொண்டிருக்கும்.

உதாரணமாக: **stdio.h** என்ற தலைப்பு கோப்பில் உள்ளீட்டு / வெளியீட்டிற்கான செயற்கூறுகளை பற்றி முன்னரே வரையறுக்கப்பட்டுள்ளது.

11.4.1 உள்ளீடு / வெளியீடு (stdio.h)

இந்த தலைப்பு கோப்பில் (உள்ளீடு / வெளியீடு) (I/O) செயற்கூறுகளான **getchar()**, **putchar()**, **gets()** மற்றும் **puts()** போன்றவற்றை முன்னரே வரையறுக்கப்பட்டுள்ளது.

11.4.1.1 getchar() மற்றும் putchar() செயற்கூறுகள்

முன்னரே வரையறுக்கப்பட்ட **getchar()** என்ற செயற்கூறினைப் பயன்படுத்தி விசைப்பலகையின் மூலம் ஒரு எழுத்தையும் உள்ளீடு செய்யலாம் மற்றும் **putchar()** என்ற செயற்கூறின் மூலம் அந்த எழுத்தையும் வெளியீடும் செய்யலாம்.

நிரல் 11.1 C++ மொழியில் ஒரு எழுத்தை உள்ளீடவும் அதை வெளியிடுவதற்கான நிரல்

```
#include<iostream>
#include<stdio.h>
using namespace std;
int main()
{
        cout<<"\n Type a Character : ";
        char ch = getchar();
        cout << "\n The entered Character is: ";
        putchar(ch);
        return 0;
}
```

வெளியீடு:

```
        Type a Character : T
        The entered Character is: T
```

11.4.1.2 gets() மற்றும் puts() செயற்கூறுகள்

gets() செயற்கூறின் மூலம் standard உள்ளீடு மூலம் ஒரு சரத்தை உள்ளீடு செய்து அதை சரத்திற்கான மாறியில் சேமித்து வைக்கலாம். puts() செயற்கூறு, gets() செயற்கூறின் மூலம் உள்ளீடாக பெற்ற சரத்தை ஒரு புதிய வரியில் வெளியிடச் செய்யும்.

நிரல் 11.2 சரத்தை உள்ளீடு மற்றும் வெளியீடு செய்வதற்கான C++ நிரல் :

```
#include<iostream>
#include<stdio.h>
using namespace std;
int main()
{
        char str[50];
        cout<<"Enter a string : ";
        gets(str);
        cout<<"You entered: "<< puts(str);
        return(0);
}
```

வெளியீடு:

```
        Enter a string : Computer Science
        You entered: Computer Science
```

11.4.1.3 குறியுறு செயற்கூறுகள் (ctype.h)

இந்த தலைப்பு கோப்பில் குறியுறுக்கு தேவையான பல்வேறு செயல்பாடுகள் வரையறுக்கப்பட்டுள்ளது. இந்த தலைப்பு கோப்பில் உள்ள பல்வேறு குறியுறு செயற்கூறுகளைக் கீழே விரிவாக விளக்கப்பட்டுள்ளது. இவ்வகை செயற்கூறுகளை நிரலில் பயன்படுத்தும் போது ctype.h என்ற தலைப்பு கோப்பை நிரலில் சேர்க்கப்பட வேண்டும்.

11.4.1.4 isalnum()

இந்த செயற்கூற்றைப் பயன்படுத்தி ஒரு குறியுறுவை ஆங்கில எழுத்தா என்று கண்டறியலாம். உள்ளீடு எண் அல்லது எழுத்தாக இருந்தால் இந்த செயற்கூறு சுழி அல்லாத மதிப்பைத் திருப்பி அனுப்பும், அல்லது சுழியம் என்ற மதிப்பைத் திருப்பி அனுப்பும்.

Syntax:

int isalnum (char c)

Example :

int r = isalnum(‘5’);

cout << isalnum('A') <<’\t’<<r;

கீழே கொடுக்கப்பட்டுள்ள கூற்றில், உள்ளீடு செய்யப்பட்ட குறியுறு ஒரு ஆங்கில எழுத்தாகவோ அல்லது எண்ணாகவோ இல்லை அதனால் n என்ற மாறியில் 0 (பூஜ்ஜியம்) என்ற மதிப்பில் இருக்கும்.

char c = '$';

int n = isalnum(c);

cout<<c;

Output:

0

நிரல் 11.3

```
#include<iostream>
#include<stdio.h>
#include<ctype.h>
using namespace std;
int main()
{
        char ch;
        int r;
        cout<<"\n Type a Character :";
        ch = getchar();
        r = isalnum(ch);
        cout<<"\nThe Return Value of isalnum(ch) is :"<<r;
}
```

வெளியீடு-1:

```
        Type a Character :A
        The Return Value of isalnum(ch) is :1
```

வெளியீடு-2:

```
        Type a Character :?
        The Return Value of isalnum(ch) is : 0
```


Chapter 11.indd 96 03/09/18 5:47 PM

11.4.1.5 isalpha()

isalpha() செயற்கூறு உள்ளீடு செய்யப்பட்ட குறியுறு ஆங்கில எழுத்தாக உள்ளதா அல்லது இல்லையா என்பதை சரிபார்க்கப் பயன்படுகிறது.

தொடரியல்: **(Syntax)**

int isalpha(char c);

இந்த செயற்கூறு உள்ளீடு செய்யப்பட்ட குறியுறு ஆங்கில எழுத்தாக இருந்தால் 1 என்ற மதிப்பைத் திருப்பி அனுப்பும் அல்லது **0** என்ற மதிப்பைத் திருப்பி அனுப்பும். கீழே கொடுக்கப்பட்டுள்ள கூற்றில் குறியுறு எழுத்தாக இல்லையெனில், வெளியீடு **0** ஆக இருக்கும்.

int n = isalpha('3');

கீழே கொடுக்கப்பட்டுள்ள கூற்றில் குறியுறு எழுத்தாக உள்ளது. அதனால் இதன் வெளியீடு 1 – ஆக இருக்கும்.

cout << isalpha('a');

நிரல் 11.4

```
#include<iostream>
#include<stdio.h>
#include<ctype.h>
using namespace std;
int main()
{
        char ch;
        cout << "\n Enter a charater: ";
        ch = getchar();
        cout<<"\n The Return Value of isalpha(ch) is :" << isalpha(ch) ;
}
```

வெளியீடு -1:

```
        Enter a charater: A
        The Return Value of isalpha(ch) is :1
```

வெளியீடு - 2:

```
        Enter a charater: 7
        The Return Value of isalpha(ch) is :0
```

11.4.2. isdigit()

உள்ளீடு செய்யப்பட்டுள்ள குறியுறு எண்ணாக உள்ளதா அல்லது இல்லையா என்பதைச் சரிபார்க்க இந்த செயற்கூறு பயன்படுகிறது. உள்ளீடு செய்யப்பட்ட குறியுறு எண்ணாக இருந்தால் இந்த செயற்கூறு 1 என்ற மதிப்பைத் திருப்பி அனுப்பும் அல்லது 0 என்ற மதிப்பைத் திருப்பி அனுப்பும்.

தொடரியல் :

int isdigit(char c);

கீழே கொடுக்கப்பட்டுள்ள நிரல் இயக்கப்படும் போது உள்ளீடு செய்யப்பட்டுள்ள குறியுறு எண்ணாக இல்லை அதனால் n என்ற மாறியில் 0 என்ற மதிப்பை இருத்தும்.

நிரல் 11.5

```
using namespace std;
#include<iostream>
#include<ctype.h>
int main()
{
        char ch;
        cout << "\n Enter a Character: ";
        cin >> ch;
        cout<<"\n The Return Value of isdigit(ch) is :" << isdigit(ch) ;
}
```

வெளியீடு -1

```
        Enter a Character: 3
        The Return Value of isdigit(ch) is :1
```

வெளியீடு -2

```
        Enter a Character: A
        The Return Value of isdigit(ch) is :0
```

11.4.2.1. islower()

இந்த செயற்கூறு உள்ளீடு செய்யப்பட்ட குறியுறு எழுத்து சிறிய எழுத்தாக உள்ளதா அல்லது இல்லையா என்று சரிபார்க்கும். உள்ளீடு செய்யப்பட்ட குறியுறு எழுத்து சிறிய எழுத்தாக இருந்தால் இந்த செயற்கூறு பூஜ்ஜியம் அல்லாத மதிப்பைத் திருப்பி அனுப்பும், இல்லையேல் 0 (பூஜ்ஜியம்) என்ற மதிப்பைத் திருப்பி அனுப்பும்.

தொடரியல் :

int islower(char c);

கீழே கொடுக்கப்பட்டுள்ள கூற்றுகளை இயக்கிய பின், உள்ளீடு செய்யப்பட்டுள்ள குறியுறு எழுத்து சிறிய எழுத்தாக உள்ளதால் 'n' என்ற மாறியில் 1 என்ற மதிப்பை இருத்தும்.

char ch = 'n';

int n = islower(ch);

ஆனால் கீழே கொடுக்கப்பட்டுள்ள கூற்றில், உள்ளீடு செய்யப்பட்டுள்ள குறியுறு எழுத்து பெரிய எழுத்தாக உள்ளதால் n என்ற மாறியின் மதிப்பு 0 ஆக இருத்தும்.

int n = islower('P');

11.4.2.2. isupper()

உள்ளீடு செய்யப்பட்டுள்ள குறியுறு எழுத்து பெரிய எழுத்தாக உள்ளதா என்று சரிபார்க்க இந்த செயற்கூறு பயன்படும். உள்ளீடு செய்யப்பட்ட குறியுறு எழுத்து பெரிய எழுத்தெனில் இந்த செயற்கூறு 1 என்ற மதிப்பைத் திருப்பி அனுப்பும் அல்லது 0 என்ற மதிப்பைத் திருப்பி அனுப்பும். கீழே கொடுக்கப்பட்டுள்ள எடுத்துக்காட்டுகளில் n என்ற மாறியில் மதிப்பு 1 என்றும் m என்ற மாறியில் மதிப்பு 0 என்றும் இருத்தும்.

Chapter 11.indd 98 03/09/18 5:47 PM

int n=isupper('A');

int m=isupper('a');

11.4.2.3. toupper()

உள்ளீடு செய்யப்பட்டுள்ள குறியுறு எழுத்து பெரிய எழுத்தாக மாற்ற இந்த செயற்கூறு பயன்படுகிறது. உள்ளீடு செய்யப்பட்ட குறியுறு எழுத்து பெரிய எழுத்தாகவே இருந்தால், வெளியீடு அதே குறியுறுவாக இருக்கும்.

தொடரியல் :

char toupper(char c);

கீழே கொடுக்கப்பட்ட கூற்று C என்ற மாறியில் 'K' என்ற மதிப்பிருத்தும்.

char c = toupper('k');

ஆனால், கீழே கொடுக்கப்பட்டுள்ள கூற்றின் வெளியீடு 'B' ஆகவே இருக்கும்.

cout <<toupper('B');

11.4.2.4. tolower()

உள்ளீடு செய்யப்பட்டுள்ள குறியுறு எழுத்தை சிறிய எழுத்தாக மாற்ற இந்த செயற்கூறு பயன்படுகிறது. உள்ளீடு செய்யப்பட்ட குறியுறு எழுத்தை சிறிய எழுத்தாக இந்த செயற்கூறு திருப்பி அனுப்பும். உள்ளீடு செய்யப்பட்ட குறியுறு ஆங்கில சிறிய எழுத்தாகவே இருந்தால், வெளியீடு அதே குறியுறுவாக இருக்கும்.

தொடரியல் :

char tolower(char c);

கீழே கொடுக்கப்பட்டுள்ள கூற்றில் c என்ற மாறியில் 'k' என்ற மதிப்பை இருத்தும்.

char c = tolower('K');

ஆனால், இந்த கூற்றில் வெளியீடு 'b' ஆகவே இருக்கும்.

cout <<tolower('b');

11.4.3 சரங்களை கையாளுதல் (string.h)

string.h (cstring) என்ற நூலக கோப்பில் குறியுறுகளின் அணியில் உள்ள சரங்களைக் கையாளுவதற்கு என பல்வேறு பொதுவான செயற்கூறுகள் உள்ளன. சரங்களைக் கையாளுவதற்கான செயற்கூறுகளைப் பயன்படுத்துவற்கு முன்பு string.h என்னும் தலைப்பு கோப்பை நிரலில் இணைத்து கொள்ள வேண்டும்.

11.4.3.1 strcpy() :

strcpy() என்ற செயற்கூறு இலக்கு (target) மற்றும் மூலம் (source) என்ற இரண்டு செயலுருபுகளை எடுத்துக் கொள்ளும். இந்த செயற்கூறு மூலத்திலுள்ள சரங்களை இலக்கு சரத்தின் நினைவகத்தில் நகல் எடுக்கும். சரத்தின் இறுதியைக் குறிக்கும் வெற்று குறியுறு (\0யையும்) இந்த செயற்கூறு நகல் எடுக்கும்.

நிரல் 11.6

```
#include <string.h>
#include <iostream>
using namespace std;
int main()
{
        char source[] = "Computer Science";
        char target[20]="target";
        cout<<"\n String in Source Before Copied :"<<source;
        cout<<"\n String in Target Before Copied :"<<target;
```

நிரல் 11.6

```
        strcpy(target,source);
        cout<<"\n String in Target After strcpy function Executed :"<<target;
        return 0;
}
```

வெளியீடு:

```
        String in Source Before Copied :Computer Science
        String in Target Before Copied :target
        String in Target After strcpy function Executed :Computer Science
```

11.4.3.2 strlen()

strlen() என்ற செயற்கூறு மூல சரத்தை அதன் செயலுருபாக எடுத்துக் கொண்டு அதன் நீளத்தை திருப்பி அனுப்பும். வெற்று குறியுறுவை (\0) சரத்தின் நீள கணக்கீட்டில் எடுத்துக்கொள்ளாது.

நிரல் 11.7

```
#include <string.h>
#include <iostream>
using namespace std;
int main()
{
        char source[ ] = "Computer Science";
        cout<<"\nGiven String is "<<source<<" its Length is "<<strlen(source);
        return 0;
}
```

வெளியீடு:

```
        Given String is Computer Science its Length is 16
```

11.4.3.3 strcmp()

strcmp() என்ற செயற்கூறு string1 மற்றும் string2 என்ற இரண்டு அளபுருக்களை எடுத்துக் கொள்ளும். இந்த செயற்கூறு string1 மற்றும் string2 உள்ளடக்கத்தை அகர வரிசையில் ஒப்பீடு செய்யும்.

strcmp() செயற்கூறு திருப்பி அனுப்பும் மதிப்புகள்:

- string1-ல் உள்ள முதல் குறியுறுவின் மதிப்பு string2 –ல் உள்ள முதல் குறியுறுவின் மதிப்பை விட அதிகமாக இருந்தால் நேர்மை மதிப்பைத் (Positive value) திருப்பி அனுப்பும். (ASCII மதிப்புகளை ஒப்பிடும்.)
- string1-ல் உள்ள முதல் குறியுறுவின் மதிப்பு string2 –ல் உள்ள முதல் குறியுறுவின் மதிப்பை விட குறைவாக இருந்தால் எதிர்ம மதிப்பைத் (Negative value) திருப்பி அனுப்பும்.

- string1 மற்றும் string2 சமம் எனில் 0 என்ற மதிப்பைத் திருப்பி அனுப்பும்.

நிரல் 11.8

```
#include <string.h>
#include <iostream>
using namespace std;
int main()
{
        char string1[] = "Computer";
        char string2[] = "Science";
        int result;
        result = strcmp(string1,string2);
        if(result==0)
        {
        cout<<"String1 : "<<string1<<" and String2 : "<<string2 <<"Are Equal";
        }
        if (result<0)
        {
        cout<<"String1 :"<<string1<<" and String2 : "<<string2 <<" Are Not Equal";
        }
}
```

வெளியீடு:

```
        String1 : Computer and String2 : Science Are Not Equal
```

11.4.3.4 strcat()

strcat() என்ற செயற்கூறு இலக்கு மற்றும் மூலம் என்ற இரு செயலுருபுகளை எடுத்துக் கொள்ளும். இந்த செயற்கூறு மூல சரத்தின் நகலை இலக்கு சரத்தின் இறுதியில் இணைக்கும்.

நிரல் 11.9

```
#include <string.h>
#include <iostream>
using namespace std;
int main()
{
        char target[50] = "Learning C++ is fun";
        char source[50] = " , easy and Very useful";
        strcat(target, source);
        cout << target ;
        return 0;
}
```

வெளியீடு:

```
        Learning C++ is fun , easy and Very useful
```


Chapter 11.indd 101 03/09/18 5:47 PM

11.4.3.5 strupr()

strupr()-இந்த செயற்கூறு உள்ளீடாக கொடுக்கப்பட்டுள்ள சரத்தை பெரிய எழுத்துக்களாக மாற்றும்.

நிரல் 11.10

```
using namespace std;
#include<iostream>
#include<ctype.h>
#include<string.h>
int main()
{
        char str1[50];
        cout<<"\nType any string in Lower case :";
        gets(str1);
        cout<<"\n Converted the Source string "<<str1<<into Upper Case is "<<strupr(str1);
        return 0;
}
```

வெளியீடு:

```
        Type any string in Lower case:computer science
Converted the Source string computer science into Upper Case is COMPUTER SCIENCE
```

11.4.3.6 strlwr()

strlwr()-இந்த செயற்கூறு உள்ளீடாக கொடுக்கப்பட்டுள்ள சரத்தை சிறிய எழுத்துக்களாக மாற்றும்.

நிரல் 11.11

```
using namespace std;
#include<iostream>
#include<ctype.h>
#include<string.h>
int main()
{
        char str1[50];
        cout<<"\nType any string in Upper case :";
        gets(str1);
        cout<<"\n Converted the Source string "<<str1<<into Lower Case is "<<strlwr(str1);
}
```

வெளியீடு:

```
        Type any string in Upper case: COMPUTER SCIENCE
Converted the Source string COMPUTER SCIENCE into lower Case is computer science
```

11.4.4 கணித செயற்கூறுகள் (math.h)

math.h என்ற தலைப்பு கோப்பு பெரும்பாலான அடிப்படை கணித செயற்கூறுகளை உள்ளடக்கி உள்ளது.

11.4.4.1 cos() செயற்கூறு

cos() செயற்கூறு ஒரு செயலுருபுவின் மதிப்பு ரேடியன்ஸ் ஆக(Radians) எடுத்துக்கொள்ளும். cos() செயற்கூறு திருப்பி அனுப்பும் மதிப்பின் அளவு [-1, 1] என்ற பரப்பில் இருக்கும். திருப்பி அனுப்பும் மதிப்பு double, float, அல்லது long double என்ற தரவினமாக இருக்கும்.

நிரல் 11.12

```
#include <iostream>
#include <math.h>
using namespace std;
int main()
{
        double x = 0.5, result;
        result = cos(x);
        cout << "COS("<<x<<")= "<<result;
}
```

வெளியீடு:

```
        COS(0.5)= 0.877583
```

11.4.4.2 sqrt() function

sqrt() செயற்கூறு உள்ளீடப்பட்ட செயலுருபின் மதிப்பிற்கான வர்க்க மூலத்தைத் திருப்பி அனுப்பும். sqrt() செயற்கூறு ஒரே ஒரு எதிர்ம எண் அல்லாத செயலுருபை மட்டுமே ஏற்கும். ஒரு எதிர்ம எண்ணை செயலுருபாக கொடுத்தால் sqrt() செயற்கூறு, செயற்களப்பகுதி (domain)பிழையை அறிவிக்கும்.

நிரல் 11.13

```
#include <iostream>
#include <math.h>
using namespace std;
int main()
{
        double x = 625, result;
        result = sqrt(x);
        cout << "sqrt("<<x<<") = "<<result;
        return 0;
}
```

வெளியீடு:

```
        sqrt(625) = 25
```


Chapter 11.indd 103 03/09/18 5:47 PM

11.4.4.3 sin() செயற்கூறு

sin() என்ற செயற்கூறு ஒரே செயலுருபுவின் மதிப்பை ரேடியன்ஸில் ஏற்கும். sin() செயற்கூறு திருப்பி அனுப்பும் மதிப்பின் பரப்பு [-1, 1]. இந்த செயற்கூறு திருப்பி அனுப்பும் மதிப்பு double, float அல்லது long double என்ற தரவுவகையிலிருக்கும் .

11.4.4.4 pow() செயற்கூறு

pow() செயற்கூறு அடித்தள(base) செயலுருபின் மேல் அடுக்குக்குறி(exponent) மதிப்பைத் திருப்பி அனுப்பும். pow() செயற்கூறின் செயலுருபின் தரவுவகை long double- ஆக இருந்தால், திருப்பி அனுப்பும் தரவின் வகை long double ஆக இருக்கும். இல்லையெனில் திருப்பி அனுப்பும் தரவுவகை double – ஆக இருக்கும். pow() செயற்கூறு இரண்டு செயலுருபுகளை ஏற்கும்.

- அடித்தளம் – அடித்தள மதிப்பு
- அடுக்குக்குறி – அடித்தள மதிப்பின் அடுக்குக்குறி

நிரல் 11.14

```
#include <iostream>
#include <math.h>
using namespace std;
int main ()
{
        double base, exponent, result;
        base = 5;
        exponent = 4;
        result = pow(base, exponent);
        cout << "pow("<<base << "^" << exponent << ") = " << result;
        double x = 25;
        result = sin(x);
        cout << "\nsin("<<x<<")= "<<result;
        return 0;
}
```

வெளியீடு:

```
        pow(5^4) = 625
        sin(25)= -0.132352
```

11.4.5 சீரற்ற (Random) எண்களை உருவாக்குதல்

C++ -ல் srand() செயற்கூறு rand() செயற்கூற்றைப் பயன்படுத்தி போலி சீரற்ற எண்களை உருவாக்கலாம். rand() செயற்கூறுக்கு 1 என்ற மதிப்பு கொடாநிலை மதிப்பாகும், இதனால் rand() செயற்கூற்றை srand() என்ற செயற்கூற்றுக்கு முன் அழைத்தால், rand() செயற்கூறு srand(1) என்ற மதிப்புடன் அழைக்கப்பட்டதாக செயல்படும். srand() செயற்கூறு குறியில்லா முழு எண்ணை அதன்

அளபுருவாக எடுத்துக்கொள்ளும், இதையே rand() செயற்கூறு அதன் தொடக்க எண்ணாக ஏற்கும். இந்த செயற்கூறு <cstdlib> அல்லது <stdlib.h> என்ற தலைப்பு கோப்பில் வரையறுக்கப்பட்டுள்ளது.

நிரல் 11.15

```
#include<iostream>
#include<cstdlib.h>
using namespace std;
int main()
{
        int random = rand(); /* No srand() calls before rand(), so seed = 1*/
        cout << "\nSeed = 1, Random number = " << random;
        srand(10);
        /* Seed = 10 */
        random = rand();
        cout << "\n\nSeed = 10, Random number = " << random;
        return 0;
}
```

வெளியீடு:

```
        Seed = 1, Random number = 41
        Seed = 10, Random number = 71
```

11.5 பயனர் வரையறுக்கப்பட்ட செயற்கூறுகள்

11.5.1 முன்னுரை

குறிப்பிட்ட பணிக்கான புதிய செயற்கூறுகளை பயனர் அவர் தேவைக்கேற்ப வரையறுக்கலாம். பயனர் உருவாக்கிய செயற்கூறுகளை பயனர் வரையறுத்த செயற்கூறுகள் என்றழைக்கப்படுகிறது. ஒரு செயற்கூறினுக்கு உள்ளீட்டு அளபுருக்களை வரையறுப்பது அவசியமில்லை, ஆனால் அவற்றை பயன்படுத்தினால் செயற்கூறை அழைக்கும் போது செயலுருபுகளுக்கான மதிப்பை அனுப்பலாம். ஒரு செயற்கூறு மதிப்பை வெளியீடாக திருப்பி அனுப்ப வேண்டும் என்பதும் அவசியமில்லை. பொதுவான செயல்பாடுகளை ஒரே மறுபயனாக்க தொகுதியில் உறை பொதியாக்கம் செய்ய செயற்கூறுகள் பயன்படும். மேலும், இந்த செயற்கூற்றுக்கு அதன் செயல்பாட்டிற்கு ஏற்ப பெயரிடுவது மிகவும் ஏற்றதாகும்.

11.5.2 செயற்கூறை வரையறுப்பது

C++-ல், ஒரு செயற்கூறை நிரலில் ஏதேனும் ஒரு பகுதியில் அதைப் பயன்படுத்தும் முன் வரையறுக்கப்பட வேண்டும். செயற்கூறு வரையறுப்பதற்கான தொடரியல்:

```
திருப்பு அனுப்பும் தரவு வகை செயற்கூறின்_பெயர்          (அளபுருக்களின் பட்டியல்)
    {
        செயற்கூறின் உடற்பகுதி
    }
```

குறிப்பு:

1. திருப்பி அனுப்பும் தரவு வகை என்பது **C++** ஏற்கும் ஏதேனும் ஒரு ஏற்கதகு தரவின வகையில் அமைக்கலாம்.

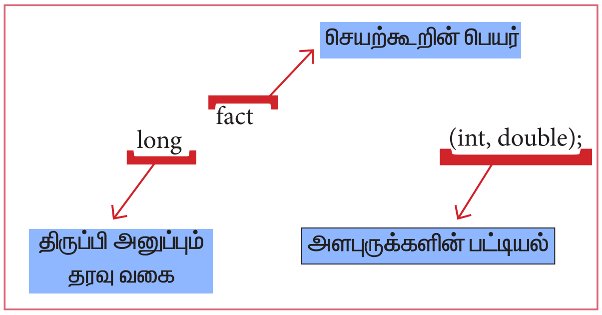
2. செயற்கூறின் பெயர் என்பது பயனர் வரையறுத்த குறிப்பெயர்கள்.

3. அளபுருக்களின் பட்டியல், கொடுப்பது அவசியமில்லை, இது மாறிகளின் பட்டியல். இதில் கொடுக்கப்பட்டுள்ள மாறியின் முன்பாக அதன் தரவினத்தை அறிவிக்கப்பட்டு, மாறிகள் காற்புள்ளியுடன் பிரிக்கப்பட்டிருக்க வேண்டும்.

4. செயற்கூறின் உடற்பகுதி- செயற்கூறின் உடல் பகுதியில் இந்த செயற்கூறு செயற்படுத்த வேண்டிய **C++** கூற்றுகளைக் கொண்டிருக்கும்.

11.5.3 செயற்கூறின் முன்வடிவு

C++ நிரல் பல செயற்கூறுகளை கொண்டிருக்கலாம். ஆனால், நிரல் செயற்பாட்டினைத் தொடங்க கண்டிப்பாக ஒரே ஒரு main() செயற்கூறு இருக்க வேண்டும். வரையறுக்கும் போது செயற்கூறுகளை நாம் அவற்றை நம் விருப்பப்படி எந்த வரிசையில் வேண்டுமானாலும் எழுதலாம். பயனர் வரையறுக்கும் செயற்கூறுகள் main() செயற்கூற்றின் முன் அல்லது பின் அமைக்கலாம். மாறியை அறிவிப்பது போன்று, ஒரு செயற்கூறையும் நிரலில் அதை பயன்படுத்தும் முன்னரே அறிவிக்கப்பட வேண்டும். செயற்கூறு அறிவிப்பு கூற்றை main() செயற்கூறின் வெளியே கொடுக்க வேண்டும்.



படம் 11.1

மேலே கொடுக்கப்பட்டுள்ள செயற்கூறின் முன் வடிவம் கீழே கொடுக்கப்பட்டுள்ள தகவல்களை நிரல்பெயர்ப்பிக்கு கொடுக்கும்.

- செயற்கூறு திருப்பி அனுப்பும் மதிப்பு **long**
- செயற்கூறின் பெயர் **fact** என்பதாகும்.
- இந்த செயற்கூறு இரண்டு செயலுருபுக்களை ஏற்கும்.

int தரவினம் இதன் முதல் செயலுருபு.

double தரவினம் இதன் இரண்டாவது செயலுருபு.

int display(int , int) //செயற்கூறின் முன்வடிவம்//

மேலே கொடுக்கப்பட்டுள்ள செயற்கூறு முன்வடிவம் இதன் திருப்பி அனுப்பும் தரவினம், பெயர் மற்றும் முறையான அளபுருக்கள் அல்லது செயலுருபுக்கள் போன்ற தகவல்களை அளிக்கிறது.

11.5.4 void கட்டளையின் பயன்

- void தரவினம் இரண்டு முக்கிய நோக்கங்கள் கொண்டது:
- இந்த செயற்கூறு எந்த மதிப்பையும் திருப்பி அனுப்பாது என்பதைக் குறிக்க.
- பொது இனச் சுட்டியை **(generic pointer)** அறிவிக்க.

குறிப்பு

செயற்கூறு எந்த மதிப்பையும் திருப்பி அனுப்பாது என்பதை நிரல்பெயர்ப்பிக்கு **void** தரவினம் குறிக்கும், மேலும் விரிவான சூழலில் **void** எந்த மதிப்பையும் ஏற்காது.

எடுத்துக்காட்டாக:

void fun(void);

மேலே குறிப்பிட்டுள்ள செயற்கூறு முன்வடிவம் நிரல்பெயர்ப்பிக்கு fun() என்ற செயற்கூறு அதை அழைக்கும் நிரலிருந்து எந்த மதிப்பையும் ஏற்றுக் கொள்ளாது. மேலும் இது அதை அழைக்கும் நிரலுக்கு எந்த மதிப்பையும் திருப்பி அனுப்பாது.

Chapter 11.indd 106 03/09/18 5:47 PM

11.5.5 செயற்கூற்றை செயல்படுத்துதல்

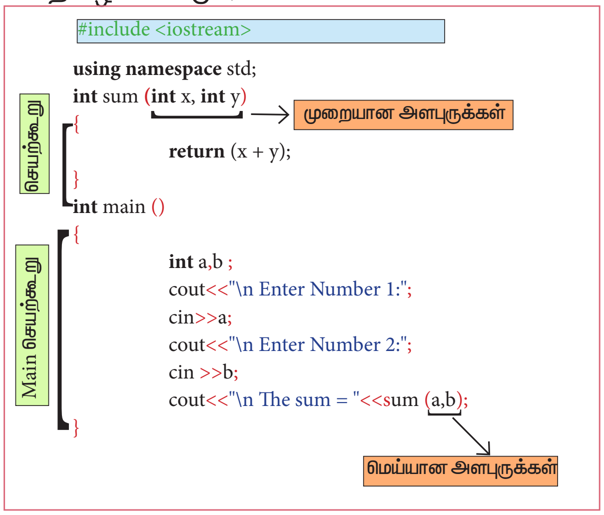
பயனர் வரையறுத்த செயற்கூறை செயல்படுத்த அதன் பெயர் மற்றும் தேவையான செயலுருபுக்கள் கொண்டு அழைக்க வேண்டும். செயற்கூறை அழைக்கப்படும் போது நிரல்பெயர்ப்பி செயற்கூறின் முன் வடிவத்தை பயன்படுத்தி செயற்கூறு சரியாக அழைக்கப்பட்டுள்ளதா என்று சரிபார்க்கும். முன்வடிவில் உள்ள செயலுருபின் தரவு வகையும் அழைப்புக் கூற்றில் உள்ள செயலுருபுக்களின் தரவு வகைகள் பொருத்தமாக இல்லையெனில், தரவு வகை மாற்றம் தானாகவே செய்ய முடியும் என்றால் நிரல்பெயர்ப்பி அதை செய்யும், இல்லையெனில் நிரல்பெயர்ப்பி இதற்கு பிழை அறிக்கை அறிவிக்கும்.

எடுத்துக்காட்டாக:

1	display()	செயலுருபு ஏற்காது மற்றும் எந்த மதிப்பையும் திருப்பி அனுப்பாத அழைப்பு செயற்கூறு.
2	display (x, y)	செயலுருபுக்களை ஏற்கும் மற்றும் எந்த மதிப்பையும் திருப்பி அனுப்பாத அழைப்பு செயற்கூறு.
3	x = display()	செயலுருபு ஏற்காது மற்றும் மதிப்பைத் திருப்பி அனுப்பும் அழைப்பு செயற்கூறு.
4	x = display (x, y)	செயலுருபுக்களை ஏற்கும் மற்றும் மதிப்பைத் திருப்பி அனுப்பும் அழைக்கும் செயற்கூறு.

14.5.5.1 முறையான அளபுருக்கள் (Formal Parameters) மற்றும் மெய்யான அளபுருக்கள் (Actual Parameters) அல்லது செயலுருபுக்கள் (Arguments)

செயலுருபுக்கள் அல்லது அளபுருக்கள் மூலமாக அழைக்கும் செயற்கூறிலிருந்து அழைக்கப்படும் செயற்கூறுக்கு மதிப்புகள் பரிமாற்றம் செய்யப்படும். வரையறுக்கப்பட்ட செயற்கூறில் மாறிகளாக பயன்படுத்தப்பட்டுள்ள அளபுருக்களை முறையான அளபுருக்கள் என்றழைக்கப்படும். அழைப்பு செயற்கூறில் உள்ள மாறிலிகள் அல்லது மாறிகள் அல்லது கோவைகளை மெய்யான அளபுருக்கள் என்றழைக்கப்படும்.



படம் 11.2 முறையான மற்றும் மெய்யான அளபுருக்கள்

11.5.5.2 முன்னியல்புச் செயலுருபுக்கள் (Default Argument)

C++ மொழியில் ஒரு செயற்கூற்றின் முன்வடிவில் உள்ள முறையான அளபுருக்களில் முன்னியல்பு மதிப்புகளை இருத்தி வைக்க முடியும். செயற்கூற்றை அழைக்கும் போது முன்னியல்பு செயலுருபு சில மதிப்புகளைத் தவிர்க்க வழிவகுக்கும். செயற்கூற்றை அழைக்கும் போது ஏதேனும் செயலுருபுகளுக்கு மதிப்பு கொடுக்காவிடில் நிரல்பெயர்ப்பி முன்னியல்பு செயலுருபுக்களின் மதிப்பைகளை அழைக்கப்பட்ட செயற்கூற்றிக்கு ஏற்கும்.

- மாறியில் தொடக்க மதிப்பிருந்தும் வடிவில் முன்னியல்பு மதிப்பு தரப்பட்டுள்ளது.

எடுத்துக்காட்டு :

void defaultvalue(int n1=10, n2=100);

- ஒரு செயற்கூறின் அழைப்புக் கூற்றில் சில செயலுருபுக்களை விட்டுவிடவோ அல்லது செயலுருபுக்கள் இல்லாமலே அழைக்கவோ முன்னியல்பு செயலுருபுக்கள் வழி வகுக்கின்றன.

எடுத்துக்காட்டு : **defaultvalue(x,y);**

defaultvalue(200,150);

defaultvalue(150);

defaultvalue(x,150);

- முன்னியல்பு மதிப்புகள் செயற்கூறு முன்வடிவில் செயலுருபுப் பட்டியலில் இறுதியாகவே இடம் பெற வேண்டும். பட்டியலில் தொடக்கத்திலோ, நடுவிலோ இடம் பெறக் கூடாது.

எடுத்துக்காட்டு :

void defaultvalue(int n1=10, n2);//invalid prototype

void defaultvalue(int n1, n2 = 10);//valid prototype

11.5.5.3 மாறிலி செயலுருபுக்கள்

const சிறப்புச் சொல்லை பயன்படுத்தி மாறிலிகளை அறிவிக்கலாம். **const** சிறப்புச் சொல் ஒரு மாறியின் மதிப்பை மாறாத தன்மை கொண்டதாக ஆகும். மாறிலியின் மாறிகள் அறிவிக்கும் போதே அதனுடைய தொடக்க மதிப்புடன் அறிவிக்க வேண்டும். **const** என்னும் பண்புணர்த்தி **(modifier)** மூலம் ஒரு மாறியில் இருத்தப்படும் மதிப்பை செயற்கூற்றின் உடல் பகுதியில் மாற்றப்பட முடியாது.

தொடரியல் :

<திருப்பி அனுப்பும் தரவு வகை> <செயற்கூறின் பெயர்> (**const** <தரவு வகை மாறி=மதிப்பு>)

எடுத்துக்காட்டு :

- int minimum(const int a=10);
- float area(const float pi=3.14, int r=5);

நிரல் 11.16

```
#include <iostream>
using namespace std;
double area(const double r,const double pi=3.14)
{
        return(pi*r*r);
}
int main ()
{
        double rad,res;
        cout<<"\nEnter Radius :";
        cin>>rad;
        res=area(rad);
        cout << "\nThe Area of Circle ="<<res;
        return 0;
}
```

வெளியீடு :

```
        Enter Radius :5
        The Area of Circle =78.5
```

“r” என்ற மாறியின் மதிப்பை r=25 என்று “area” செயற்கூறின் உடல் பகுதியில் மாற்றியமைத்தால், நிரல்பெயர்ப்பி “assignment of read-only parameter 'r'” என்ற பிழையை அறிவிக்கும்.

```
double area(const double r, const double pi=3.14)
{
```


Chapter 11.indd 108 03/09/18 5:47 PM

```
        r=25;
        return(pi*r*r);
}
```

11.6 செயற்கூற்றை அழைப்பதற்கான வழிமுறைகள்

C++ மொழியில் செயலுருபுக்களை செயற்கூறுக்கு இரு வழிகளில் அனுப்பலாம். செயலுருபுக்களை அனுப்பும் முறையைப் பொருத்து, செயற்கூற்றை அழைக்கும் வழிமுறைகள் மதிப்பு மூலம் அழைத்தல் (Call by Value)மற்றும் குறிப்பு மூலம் அல்லது முகவரி மூலம் அழைத்தல் (Call by Reference or Address) என்று இரு வகைப்படுத்தப்படும்.

11.6.1 மதிப்பு மூலம் அழைத்தல் முறை

இந்த முறையில் மெய்யான அளபுருவின் மதிப்பை முறையான அளபுருவில் நகலெடுக்கும். இந்த முறையில் முறையான அளபுருவின் மதிப்பில் ஏதேனும் மாற்றங்கள் செய்தால் அது மெய்யான அளபுருவின் மதிப்பில் பிரதிபலிப்பதில்லை.

நிரல் 11.17

```
#include<iostream>
using namespace std;
void display(int x)
{
        int a=x*x;
        cout<<"\n\nThe Value inside display function (a * a):"<<a;
}
int main()
{
          int a,b;
        cout<<"\nExample : Function call by value:";
          cout<<"\n\nEnter the Value for A :";
          cin>>a;
        display(a);
        cout<<"\n\nThe Value inside main function "<<a;
        return(0);
}
```

வெளியீடு :

```
        Example : Function call by value
        Enter the Value for A : 5
        The Value inside display function (a * a) : 25
        The Value inside main function 5
```

11.6.2 குறிப்பு மூலம் அழைத்தல் முறை

இந்த முறையில் மெய்யான அளபுருவின் குறிப்பை அல்லது முகவரியை முறையான அளபுருவில் நகலெடுக்கும். முகவரியின் மூலம் அழைப்பதால் முறையான அளபுருவின் மதிப்பில் ஏதேனும் மாற்றம் செய்தால் மெய்யான அளபுருவில் அந்த மாற்றம் பிரதிபலிக்கும்.

நிரல் 11.18

```
#include<iostream>
using namespace std;
void display(int &x) //passing address of a//
{
         x=x*x;
        cout<<"\n\nThe Value inside display function (n1 x n1) :"<<x ;
 }
int main()
{
int n1;
cout<<"\nEnter the Value for N1 :";
cin>>n1;
cout<<"\nThe Value of N1 is inside main function Before passing : "<< n1;
display(n1);cout<<"\nThe Value of N1 is inside main function After passing (n1 x n1)  : "<< n1;
return(0);
}
```

வெளியீடு :

```
Enter the Value for N1 :45
The Value of N1 is inside main function Before passing : 45
The Value inside display function (n1 x n1) :2025
The Value of N1 is inside main function After passing (n1 x n1)  : 2025
```

display() செயற்கூற்றின் தலைப்புப் பகுதியில் உள்ள மாற்றத்தை கவனிக்கவும். & என்ற குறியுடன் x என்ற அளபுருவை அறிவிக்கப்பட்டு, இதை குறிப்பு மாறி என்பதை உணர்த்தும் இதனால் இந்த செயற்கூறை அழைக்கப்படும்போது குறிப்பு மூலம் அழைக்கப்பட வேண்டும். இதனால் display() செயற்கூறை அழைக்கும் போது num1 என்ற மாறியின் முகவரியை x மாறி பகிர்ந்து கொள்ளும். main() செயற்கூறில் num1 என்ற மாறியின் பெயரும் display() செயற்கூறில் x என்ற மாறியின் பெயரும் ஒரே சேமிப்பு பகுதியைக் குறிக்கிறது. இதனால் x மாறியின் மதிப்பை மாற்றியமைத்தால் அந்த மாற்றம் num1 மாறியின் மதிப்பை உண்மையில் மாற்றும்.

11.6.3 Inline செயற்கூறு

ஒரு செயற்கூறை அழைக்கும் கூற்று நிரல்பெயர்ப்பியை செயற்கூறுக்கு (வரையறுக்கப்பட்ட செயற்கூறுகள் அடுக்கங்களில் (STACKS) சேமிக்கப்படும்) தாவச்செய்து மற்றும் செயற்கூறு செயல்பாட்டின் பின் அழைப்புக்கூறுக்கு அடுத்திருக்கும் கட்டளைக்குத் தாவச்செய்கிறது. இதனால் நிரலின் இயக்க நேரத்தின் வேகம் குறையும். சிறிய செயற்கூறின் வரையறுப்புக்கு அடுக்கங்களைப் பயன்படுத்துவதைக் குறைக்க Inline செயற்கூறுகள் பயன்படுத்தலாம்.

inline செயற்கூறு மூல நிரலில் சாதாரணச் செயற்கூறு போன்றே தோற்றமளிக்கும். ஆனால் செயற்கூறின் கட்டளைகள் முழுமையும் அழைப்புக் கூற்றுக்குப் பதிலாக அப்படியே நிரலில்

Chapter 11.indd 110 03/09/18 5:47 PM

செருகப்பட்டுவிடும். **Inline** சிறப்புச் சொல்லை செயற்கூறின் தலைப்பில் இணைத்து அந்த செயற்கூறை **inline** செயற்கூற்றாக மாற்ற முடியும்.

தொடரியல் :

inline <திருப்பி அனுப்பும் வகை > < செயற்கூறின் பெயர் (தரவு வகை அளபுருக்களின் பெயர் 1, … தரவு வகை அளபுருக்களின் பெயர் N - வரை)>

inline செயற்கூற்றுகளின் நன்மைகள்

- **inline** செயற்கூறுகள் வேகமாக செயல்படும். ஆனால் அதிக நினைவக இடத்தை எடுத்துக் கொள்ளும்.
- அடுக்கங்களை பயன்படுத்தும் போது உள்ள சிக்கல்பாட்டினைக் குறைக்கிறது.

நிரல் 11.19

```
#include <iostream>
using namespace std;
inline float simpleinterest(float p1,float n1, float r1)
{
        float si1=(p1*n1*r1)/100;
        return(si1);
}
int main ()
{
        float  si,p,n,r;
        cout<<"\nEnter the Principle Amount Rs. :";
        cin>>p;
        cout<<"\nEnter the Number of Years      :";
        cin>>n;
        cout<<"\nEnter the Rate of Interest     :";
        cin>>r;
        si=simpleinterest(p,n,r);
        cout << "\nThe Simple Interest = Rs."<<si;
        return 0;
}
```

வெளியீடு:

```
        Enter the Principle Amount Rs. :60000
        Enter the Number of Years      :10
        Enter the Rate of Interest     :5
        The Simple Interest = Rs.30000
```

மேலே கொடுக்கப்பட்டுள்ள நிரலில் உள்ள செயற்கூறின் வரையறுப்பு பொதுவான வடிவில் இருந்தாலும் நிரல்பெயர்ப்பிக்கு பின் செயற்கூறை இயக்கும் போது (p1*n1*r1)/100 என்ற செயற்கூறின் கூற்று

si=simpleinterest(p,n,r); என்று அழைக்கும் கூற்றை si= (p1*n1*r1)/100; என்று மாற்றியமைக்கும்.

11.7 பயனர் வரையறுத்த செயற்கூற்றுகளை அறிவிக்கும் போது பயன்படுத்தப்படும் பல்வேறு வடிவங்கள்

11.7.1 மதிப்பை திருப்பி அனுப்பாத மற்றும் அளபுருக்களை ஏற்காத செயற்கூறு

கீழே கொடுக்கப்பட்டுள்ள நிரல் மதிப்பைத் திருப்பி அனுப்பாத மற்றும் அளபுருக்களை ஏற்காத செயற்கூறிற்கான ஒரு எடுத்துக்காட்டு.

display() என்பது செயற்கூறின் பெயர், இதன் திருப்பி அனுப்பும் தரவினம் void மற்றும் இந்த செயற்கூறு எந்த அளபுருவையும் ஏற்காது.

நிரல் 11.20

```
#include<iostream>
using namespace std;
void display()
{
        cout<<"First C++ Program with Function";
}
int main()
{
        display(); // Function calling statement//
        return(0);
}
```

வெளியீடு :

```
        First C++ Program with Function
```

11.7.2 திருப்பி அனுப்பும் மதிப்பு மற்றும் அளபுருக்களை ஏற்காத செயற்கூறு

display() என்ற செயற்கூறின் திருப்பி அனுப்பும் தரவினம் int மற்றும் இந்த செயற்கூறு அளபுருவையும் ஏற்காது. return செயற்கூறு அழைப்பு செயற்கூறுக்கு மதிப்பைத் திருப்பி அனுப்பும் மற்றும் நிரலின் கட்டுப்பாட்டை மீண்டும் அழைப்புக் கூற்றுக்கு திருப்பி அனுப்பும்.

நிரல் 11.21

```
#include<iostream>
using namespace std;
int display()
{
        int a, b, s;
        cout<<"Enter 2 numbers: ";
        cin>>a>>b;
        s=a+b;
        return s;
        }
```


Chapter 11.indd 112 03/09/18 5:47 PM

```
int main()
{
        int m=display();
       cout<<"\nThe Sum="<<m;
       return(0);
}
```

வெளியீடு :

```
       Enter 2 numbers: 10 30
       The Sum=40
```

11.7.3 மதிப்பை திருப்பி அனுப்பாத மற்றும் அளபுருக்களை ஏற்கும் செயற்கூறு

display() என்ற செயற்கூறின் திருப்பி அனுப்பும் தரவினம் void, மேலும் இது x மற்றும் y என்ற இரண்டு அளபுருக்கள் அல்லது செயலுருபுக்களின் மதிப்புகளை ஏற்கும். return கூற்று கட்டுப்பாட்டை அழைப்பு கூற்றுக்குத் திருப்பி அனுப்பும்.

நிரல் 11.22

```
#include<iostream>
using namespace std;
void display(int x, int y)
{
        int s=x+y;
        cout<<"The Sum of Passed Values: "<<s;
}
int main()
{
        int a,b;
        cout<<"\nEnter the First Number    :";
        cin>>a;
        cout<<"\nEnter the Second Number  :";
          cin>>b;
          display(a,b);
        return(0);
}
```

வெளியீடு :

```
        Enter the First Number    :50
        Enter the Second Number  :45
        The Sum of Passed Values: 95
```

11.7.4 மதிப்பை திருப்பி அனுப்பும் மற்றும் அளபுருவை ஏற்கும் செயற்கூறு

display(), என்ற செயற்கூறு **int** என்ற மதிப்பைத் திருப்பி அனுப்பும், மேலும் **x** மற்றும் **y** என்ற இரண்டு அளபுருக்கள் அல்லது செயலுருபுக்களில் மதிப்புகளை ஏற்கும். **return** கூற்று கட்டுப்பாட்டை அழைப்பு கூற்றிக்குத் திருப்பி அனுப்பும்.

Chapter 11.indd 113 03/09/18 5:47 PM

நிரல் 11.23

```
#include<iostream>
using namespace std;
int display(int x, int y)
{
        int s=x+y;
        return s;
}
int main()
{
        int a,b;
        cout<<"\nEnter the First Number  :";
        cin>>a;
        cout<<"\nEnter the Second Number  :";
        cin>>b;
        int s=display(a,b);
        cout<<"\nExample:Function with Return Value and  with Arguments";
        cout<<"\nThe Sum of Passed Values: "<<s;
        return(0);
}
```

வெளியீடு :

```
Enter the First Number  :45
Enter the Second Number  :67
Example: Function with Return Value and  with Arguments
The Sum of Passed Values: 112
```

11.8 கட்டுப்பாட்டை செயற்கூறிலிருந்து திருப்பி அனுப்புதல்

return கூற்றை பயன்படுத்தி கட்டுப்பாட்டை செயற்கூறிலிருந்து திரும்பப் பெறலாம்.

return கூற்று செயற்கூறின் இயக்கத்தை நிறுத்தி கட்டுப்பாட்டை அழைத்து செயற்கூறுக்கு திருப்பி அனுப்பும். செயற்கூற்றின் பகுதியில் எங்கு return கூற்று இயக்கப்படுகிறதோ, உடனே அந்த இடத்தில் செயற்கூறின் செயல்பாடு நிறுத்தப்பட்டு, கட்டுப்பாடு அழைப்பு கூற்றுக்குத் திரும்பும்.

11.8.1 return கூற்று

செயற்கூறிலிருந்து கட்டுப்பாட்டைத் திரும்பப் பெற **return** கூற்று பயன்படுகிறது. இதை தாவும் கூற்றாக வகைப்படுத்தலாம். ஏனென்றால் இந்த கூற்றை இயக்கியவுடன் செயற்கூறின் இயக்கம் நிறுத்தப்பட்டு, கட்டுப்பாடு அழைக்கும் கூற்றுக்கு திரும்ப அனுப்பலாம். **return** கூற்று அதனோடு தொடர்புடைய மதிப்பு உடன் அல்லது மதிப்பு இல்லாமலோ இருக்கலாம். மதிப்பு உடன் உள்ள **return** கூற்று அழைப்பு கூற்றுக்கு அந்த மதிப்பைத் திருப்பி அனுப்பும். **void** செயற்கூறிற்க்கும் அளபுரு ஏற்காத **return** கூற்றைப் பயன்படுத்தி அந்த செயற்கூற்றின் இயக்கத்தை நிறுத்தலாம்.

தொடரியல்:

return கேவை/மாறி;

எடுத்துக்காட்டு :

```
return(a+b);   return(a);

return; // to terminate the function
```

11.8.2 திருப்பி அனுப்பும் மதிப்புகள்:

மதிப்பினைத் திருப்பி அனுப்பாத செயற்கூற்றினை **void** என்று அறிவிக்கலாம். எந்த தரவினத்தையும் வெளிப்படையாக அறிவிக்கவில்லையெனில் செயற்கூறின் திருப்பி அனுப்பும் தரவினம் **int** என்று கருதும்.

எடுத்துக்காட்டு :

```
int add (int, int);

add (int, int);
```

இவ்விரு முன்வடிவங்களும், int என்ற மதிப்பை திருப்பி அனுப்பும், ஏனெனில் C++ மொழியில் செயற்கூறு திருப்பி அனுப்பும் மதிப்பு தானமைவாகவே int ஆக இருக்கும். கீழ்காணும் எடுத்துக்காட்டுகளை காண்க.

வரிசை எண்	செயற்கூறு முன்வடிவு	திருப்பியனுப்பும் தரவினம்
1	int sum (int, float)	int
2	float area(float, float)	float
3	char result()	char
4	double fact(int n)	double

முழு எண் அல்லாத மதிப்புகளை திருப்பி அனுப்புதல்

அழைப்பு கூற்றிற்கு சரத்தை கூட திருப்பி அனுப்ப முடியும்.

நிரல் 11.24

```
#include<iostream>
#include<string.h>
using namespace std;
char *display()
{
        return (“chennai”);
}
```

```
int main()
{
        char s[50];
        strcpy(s,display());
        cout<<"\nExample:Function with Non Integer Return"<<s;
        return(0);}
```

வெளியீடு :

Example: Function with Non Integer Return Chennai

11.8.3 குறிப்பு மூலம் திருப்பி அனுப்புதல்

நிரல் 11.25

```
#include<iostream>
using namespace std;
int main()
{
        int n1=150;
        int &n1ref=n1;
        cout<<"\nThe Value of N1 = "<<n1<<" and n1Reference = "<<n1ref;
        n1ref++;
        cout<<"\nAfter n1 increased the Value of N1 = "<<n1;
        cout<<" and n1Reference = "<<n1ref;
        return(0);
}
```

வெளியீடு:

The Value of N1 = 150 and n1Reference = 150
After n1 increased the Value of N1 = 151 and n1Reference = 151

குறிப்பு

n1ref என்ற மாறி n1 மாறிக்கு மாற்றுப் பெயராக செயல்படுகிறது. இதனால் n1ref மாறியின் மதிப்பை மாற்றி அமைத்தால் தானாகவே n1 மாறியின் மதிப்பும் மாறும்.

n1 மற்றும் n1ref என்ற இரண்டு மாறிகளும் ஒரே நினைவகம் அல்லது குறிப்பைப் பகிர்ந்து கொள்ளும்.

11.9 தற்சுழற்சி செயற்கூறு (Recursive Function)

ஒரு செயற்கூறு தன்னைத் தானே அழைத்துக் கொண்டால் அதை தற்சுழற்சி செயற்கூறு என்று அறியப்படும். இந்த நுட்பத்தை தற்சுழற்சி முறை என்றழைக்கப்படும்.

எடுத்துக்காட்டு 1: தற்சுழற்சி முறையில் ஒரு எண்ணின் காரணியை கணக்கிடுக.

நிரல் 11.26

```
#include <iostream>
using namespace std;
int factorial(int); // Function prototype //
int main()
{
        int no;
        cout<<"\nEnter a number to find its factorial: ";
        cin >> no;
        cout << "\nFactorial of Number " << no <<" = " << factorial(no);
        return 0;
}
int factorial(int m)
{
        if (m > 1)
        {
        return m*factorial(m-1);
        }
        else
        {
        return 1;
}
}
```

வெளியீடு :

```
        Enter a number to find its factorial: 5
        Factorial of Number 5 = 120
```

குறிப்பு: main() செயற்கூறின் பின்பு factorial() செயற்கூறை எழுதியிருப்பதால் இந்த செயற்கூறின் முன் வடிவம் கொடுப்பது அவசியமானது.

எடுத்துக்காட்டு 2: தற்சுழற்சி முறையில் ஒரு எண்ணின் மிக அதிகமான பொதுவான காரணியை HCF கணக்கிடுக.

Chapter 11.indd 117 03/09/18 5:47 PM

நிரல் 11.27

```
#include <iostream>
using namespace std;
//Function to find HCF //
int hcf(int n1, int n2)
{
        if (n2 != 0)
        return hcf(n2, n1 % n2);
        else
        return n1;
}
int main()
{
        int num1, num2;
        cout << "Enter two positive integers: ";
        cin >> num1 >> num2;
        cout << "Highst Common Factor (HCF) of " << num1;
        cout<< " & " << num2 << " is: " << hcf(num1, num2);
        return 0;
}
```

வெளியீடு:

```
        Enter two positive integers: 350 100
        Highst Common Factor (HCF) of 350 & 100 is: 50
```

குறிப்பு: hcf() என்ற செயற்கூறு main() செயற்கூற்றின் முன் எழுதப்பட்டிருப்பதால் இந்த hcf() செயற்கூற்றின் முன்வடிவம் எழுதுவது அவசியமில்லை.

11.10 மாறிகளின் வரையெல்லை விதிமுறைகள் (Scope rules of Variables)

வரையெல்லை என்பது ஒரு மாறியின் அணுகியல்பைக் குறிக்கிறது. C++ மொழியில் நான்கு வகையான வரையெல்லைகள் உள்ளன. அவை உள்ளமை வரையெல்லை (Local scope), செயற்கூறு வரையெல்லை (Function scope), கோப்பு வரையெல்லை (File scope) மற்றும் இனக்குழு வரையெல்லை (Class scope).

11.10.1 முன்னுரை

வரையெல்லை என்பது ஒரு மாறி செயல்படும் வரம்பெல்லை அல்லது அதன் வாழ்நாள் வரையாகும். மேலும் இதை விவரிக்கும் போது மாறிகளை நான்கு இடங்களில் அறிவிக்கலாம்.

- ஒரு தொகுதிக்குள் அறிவிக்கும்போது அவற்றை உள்ளமை மாறிகள் என்றழைக்கப்படும்.
- செயல்கூறின் உள்ளே அறிவித்தால் அவற்றை செயல்கூறு மாறிகள் என்றழைக்கப்படும்.
- எல்லா செயற்கூறுக்கும் வெளியே அறிவித்தால், அவற்றை பொதுமையான (Global) மாறிகள் என்றழைக்கப்படும்.

Chapter 11.indd 118 03/09/18 5:47 PM

- இனக்குழுவில் உள்ளே அறிவித்தால் அவற்றை இனக்குழு மாறிகள் அல்லது தரவு உறுப்புகள் (data members) என்று அழைக்கப்படும்.

11.10.2 உண்மை வரையெல்லை:

- உள்ளமை மாறி, ஒரு தொகுதிக்குள் (Block) வரையறுக்கப்படுகிறது. ஒரு தொகுதியில் உள்ள நிரல் { } என்ற அடைப்புக்குறிக்குள் இருக்கும்.
- ஒரு உள்ளமை மாறியின் வரையெல்லை அது வரையறுக்கப்பட்டுள்ள தொகுதிக்குள் மட்டுமே இருக்கும்.
- ஓர் உள்ளமை மாறியை அது அறிவிக்கப்பட்டுள்ள தொகுதிக்கு வெளியிலிருந்து அணுக முடியாது.
- நிரலின் கட்டுப்பாடு ஒரு கட்டளைத் தொகுதிக்குள் நுழையும் போது, அதன் உள்ளமை மாறிகள் உருவாக்கப்படுகின்றன. வெளியேறும் போது அவை அழிக்கப்படுகின்றன.

நிரல் 11.28 (a)

```
//Demo to test Local Scope//
#include<iostream>s
using namespace std;
int main ( )
{
int a, b ;
a = 10;
b = 20;
if (a > b)
{
int temp; //local to this if block//
temp = a;
a = b;
b = temp;
}
cout <<"\n Descending order .... \n";
cout <<a <<"\t"<<b;
return(0);
}
```

வெளியீடு:

```
Descending order ....
10    20
```

நிரல் 11.28 (ஆ)

```
//Demo to test Local Scope//
#include<iostream>s
using namespace std;
int main ( )
{
int a, b ;
a = 10;
b = 20;
if (a > b)
{
int temp; //local to this if block//
temp = a;
a = b;
b = temp;
}
cout <<temp;
return(0);
}
```

குறிப்பு **In function 'int main()':**

[Error] 'temp' was not declared in this scope

மேலே கொடுக்கப்பட்டுள்ள நிரலைச் செயல்படுத்தும் பொழுது, நிரல் பெயர்ப்பி ஒரு பிழை செய்தியைக் கொடுக்கிறது. temp என்னும் மாறியை அணுகமுடியாது. ஏனெனில் அதன் வாழ்நாள் இயங்கு நிலையில் அதன் தொகுதியில் வாழ்நாள் வரைக்கும் இருக்கும் தொகுதி நிறைவேற்றப்பட்டபின் உள்ள tamil மாறி வாழ்நாள் முடிந்து விடும்.

11.10.3 கோப்பு வரையெல்லை:

- செயற்கூறினுள் அறிவிக்கப்பட்ட மாறியின் வரையெல்லை அந்த செயற்கூறின் தொகுதி மற்றும் துணை தொகுதி வரை உள்ளது.
- மாறியின் வாழ்நாள் செயற்கூறு தொகுதியின் வாழ்நாள்வரைக்கும் இருக்கும் முறையான அளபுருக்களின் வரையெல்லை செயற்கூறின் வரையெல்லை ஆகும்.
- கோப்பு வரையெல்லை மாறியை பொதுமை மாறிகள் என்றழைக்கப்படும்.

Chapter 11.indd 120 03/09/18 5:47 PM

நிரல் 11.29 (b)

```
//Demo to test Function Scope//
#include<iostream>
using namespace std;
void add(int x, int y)
{
        int m=x+y; //'m' declared within function add()//
        cout<<"\nThe Sum = "<<m;
}
int main ( )
{
int a, b ;
a = 10;
b = 20;
add(a,b);
cout<<m; //'m' declared within function add()//
return(0);
}
```

வெளியீடு:

```
        The Sum = 30
```

குறிப்பு : **In function 'int main()':**

[Error] 'm' was not declared in this scope

11.10.4 கோப்பு வரையெல்லை:

- அனைத்துக் கட்டளைத் தொகுதிகளுக்கும் செயற்கூறுகளுக்கும் மேலாக (குறிப்பாக main ( ) செயற்கூறினுக்கு மேலே) அறிவிக்கப்படும் மாறி, கோப்பு வரையெல்லை கொண்டதாகும். கோப்பு வரையெல்லை அந்த நிரலின் முழுமையும் விரிகிறது. அதன் வாழ்நாள் அந்த நிரல் செயல்பட்டு முடியும் வரை நீடிக்கும்.
- கோப்பு வரையெல்லை மாறியை பொதுமை மாறிகள் என்றழைக்கப்படும்.

நிரல் 11.30

```
//Demo to test File or global Scope//
#include<iostream>
using namespace std;
int file_var=20; //Declared within File//
void add(int x, int y)
{
        int m=x+y+file_var;
        cout<<"\n The Sum = "<<m;
}
int main ( )
{
int a, b ;
a = 10;
b = 20;
add(a,b);
cout<<"\nThe File Variable = "<<file_var;
return(0);
}
```

வெளியீடு:

```
        The Sum = 50
        The File Variable =20
```

11.10.5 இனக்குழு வரையெல்லை:

- பயனர்கள் புதிய தரவினங்களை உருவாக்கவும், நடைமுறைப் படுத்தவும் ஒரு புதிய வழியை இனக்குழு திறக்கிறது. வேறுபட்ட இனத்தரவுகளை ஒன்றாகச் சேர்த்து வைக்க இனக்குழுக்கள் ஒரு புதிய வழிமுறையை வழங்குகின்றன.
- தரவு உறுப்புகள் தரவு மாறிகள் என்று அழைக்கப்படும், இவை இனக்குழுவின் பண்புக்கூறுகளை உணர்த்தும்.

class student
{
private :
int mark1, mark2, total;
};
	student என்ற இனக்குழுவில் mark1, mark2 மற்றும் total என்பவை தரவு மாறிகள். இந்த தரவு மாறிகளின் வரையெல்லை student இனக்குழுவின் வரையெல்லையாகும்.

Chapter 11.indd 122 03/09/18 5:47 PM

குறிப்பு

"Classes and Object" என்ற அதிகாரத்தில் இனக்குழு வரையெல்லை பற்றி விரிவாக தெரிந்து கொள்ளலாம்.

11.10.6 வரையெல்லை தெளிவுபடுத்தும் செயற்குறி (Scope resolution operator)

வரையெல்லை தெளிவுபடுத்தும் செயற்குறி ஒரு மாறியின் மறைந்து கிடக்கும் வரையெல்லையை வெளிக்கொணரும். வரையெல்லை தெளிவுபடுத்தும் செயற்குறி :: இதற்கு பயன்படுகிறது.

- பொதுமையான மாறியும் உள்ளமை மாறியும் ஒரே பெயரை கொண்டிருப்பதால், பொதுமையான மாறியை இயக்க வரையெல்லை தெளிவுபடுத்தும் செயற்குறி பயன்படுத்தும் ஒரு எடுத்துக்காட்டு.

நிரல் 12.31

```
// Program to show that we can access a global variable
// using scope resolution operator :: when there is a local
// variable with same name //
#include<iostream>
using namespace std;
int x=45;  // Global Variable x
int main()
{
  int x = 10; // Local Variable x
cout << "\nValue of global x is " << ::x;
  cout << "\n\nValue of local x is " << x;
  return 0;
}
```

வெளியீடு:

```
Value of global x is 45
Value of local x is 10
```

ஆய்வு அறிக்கை :

1. வர்க்க மூலம் (square root) , அடுக்கின் மதிப்பு (power values), tan, கன மூலம் (cube root) போன்றவற்றைக் கண்டறிய செயற்கூறுகளைப் பயன்படுத்தி நிரலை எழுதுக.
2. ஐந்து மாணவர்களின் பெயர்களை அவர்களின் தலைப்பு எழுத்தை இறுதியில் அமையுமாறு உள்ளீடாக செய்க, பெயரை ஆங்கில சிறிய மற்றும் பெரிய எழுத்துக்களாக வெளியீடாக செய்யவும். மேலும் ஒவ்வொரு பெயருக்கும் உள்ள எழுத்துக்களின் எண்ணிக்கை வெளியீடாக பெற உரிய நிரலை எழுதுக.
3. காரணிப்படுத்துதல் (factorial), பகா எண்(prime number), ஆம்ஸ்டார்ங் எண்கள் (Armstrong numbers) போன்றவை கண்டறிய செயற்கூறுகளை பயன்படுத்தி நிரலை எழுதுக.
4. ஒருவரின் பெயர் மற்றும் பாலினம் உள்ளீடாக பெற்று திரு/ திருமதி என்ற சொல்லை பெயருடன் இணைத்து வெளியிடுவதற்கு உரிய நிரலை எழுதுக.

Chapter 11.indd 123 03/09/18 5:47 PM

நினைவில் கொள்க

- ஒரு பெரிய நிரலை சிறு சிறு பகுதிகளாக பிரிப்பதையே செயற்கூறுகளாகும்.
- செயற்கூறுகள் முன் வரையறுக்கப்பட்ட அல்லது உள்ளிணைந்த அல்லது நூலக செயற்கூறுகள் மற்றும் பயனர் வரையறுத்த செயற்கூறுகள் என வகைப்படுத்தலாம்.
- பயனர் வரையறுத்த செயற்கூறுகள் பயனரால் உருவாக்கப்படும்.
- void செயற்கூறு எந்த மதிப்பையும் செயற்கூறு திருப்பி அனுப்பாது என்று நிரல்பெயர்ப்பிக்கு தெரிவிக்கும்.
- return கூற்று அழைப்பு செயற்கூற்றுக்கு ஒரு மதிப்பை திருப்பி அனுப்பியும், மேலும் அழைக்கும் செயற்கூற்றிற்கு கட்டுப்பாட்டை திருப்பி அனுப்பும்.
- C++ மொழியில் தானமைவாக திருப்பி அனுப்பும் தரவு வகை int தரவினமாகும்.
- தன்னைத் தானே அழைத்துக் கொள்ளும் செயற்கூறுகளை தற்சுழற்சி செயற்கூறு என்றழைக்கப்படும்.
- வரையெல்லை என்பது ஒரு மாறியின் அணுகியல்பைக் குறிக்கும்.
- உள்ளமை வரையெல்லை, செயற்கூறு வரையெல்லை, கோப்பு வரையெல்லை மற்றும் இனக்குழு வரையெல்லை என நான்கு வகையான வரையெல்லைகள் உள்ளன.
- வரையெல்லை செயற்குறி (::) ஒரு மாறியின் மறைக்கப்பட்ட வரையெல்லையை தெரிவிக்கும்.

மதிப்பீடு

N1H9CG

பகுதி அ

I சரியான விடை யை தேர்வு செய்யவும் :

1. இவற்றுள் எந்த தலைப்பு கோப்பு நிலையான I/O விற்கான முன்வரையறுக்கப்பட்ட செயற்கூறுகளை வரையறுக்கும் ?

   அ) stdio.h   ஆ) math.h   இ) string.h   ஈ) ctype.h

2. ஒரு குறியுறுவை எழுத்து மற்றும் எண் வகையா அல்லது இல்லையா என்பதை சரிபார்க்க உதவும் செயற்கூறு எது?

   அ) isalpha()   ஆ) isdigit()   இ) isalnum()   ஈ) islower()

3. நிரலின் செயலாக்கம் எந்த செயற்கூறிலிருந்து தொடங்கும் ?

   அ) isalpha()   ஆ) isdigit()   இ) main()   ஈ) islower()

4. இவற்றுள் எந்த செயற்கூறு ஒரு மதிப்பைத் திருப்பி அனுப்பி மற்றும் செயலுருபுகளை ஏற்காத செயற்கூறு ஆகும்?

   அ) x=display(int, int)   ஆ) x=display()   இ) y=display(float)   ஈ) display(int)

Chapter 11.indd 124   03/09/18 5:47 PM

5. add(int, int); என்ற செயற்கூற்றின் முன்வடிவின் திருப்பி அனுப்பும் தரவினத்தின் வகை யாது?

அ) int ஆ) float இ) char ஈ) double

6. இவற்றுள் எது வரையெல்லை செயற்குறியாகும் ?

அ) > ஆ) & இ) % ஈ) ::

பகுதி - ஆ

குறு வினாக்கள் (2 மதிப்பெண்கள்) :

1. செயற்கூறுகள் - வரையறை
2. strlen() செயற்கூறை பற்றி எழுதுக
3. void தரவு வகையின் முக்கியத்துவங்கள் என்ன?
4. அளபுரு என்றால் என்ன ? அதன் வகைகளை பட்டியலிடுக
5. உள்ளமை வரையெல்லை பற்றி சிறுகுறிப்பு வரைக.

பகுதி - இ

சிறு வினாக்கள் (3 மதிப்பெண்கள்) :

1. உள்ளிணைந்த செயற்கூறுகள் என்றால் என்ன ?
2. suppr() மற்றும் toupper() செயற்கூறுகளின் வேறுபாடுகள் யாவை?
3. strcmp() செயற்கூறு பற்றி குறிப்பு வரைக.
4. C++ மொழியில் உள்ள pow() செயற்கூறு பற்றி சிறுகுறிப்பு வரைக.
5. செயற்கூறு முன்வடிவம் நிரல்பெயர்ப்பிக்கு எந்த தகவலை வழங்கும் ?
6. முன்னிலைப்பு செயலுருபுக்கள் என்றால் என்ன ? எடுத்துக்காட்டு தருக.

பகுதி - ஈ

பெரு வினாக்கள் (5 மதிப்பெண்கள்) :

1. மதிப்பு மூலம் அழைத்தல் முறையை தகுந்த எடுத்துக்காட்டுடன் விளக்குக.
2. தற்சுழற்சி என்றால் என்ன ? தற்சுழற்சி முறையில் ஒரு எண்ணிற்கான மிகப்பெரிய பொதுவான காரணியை கணக்கிட ஒரு நிரலை எழுதுக.
3. செயற்கூறு மதிப்பை திருப்பி அனுப்பும் பல்வேறு வடிவங்களை எடுத்துக்காட்டுடன் விளக்குக.
4. மாறியின் வரையெல்லை விதிமுறைகளை எடுத்துக்காட்டுடன் விளக்குக.
5. ஒரு முழு எண்ணை உள்ளீட்டு அதை தலைகீழாக மாற்றம் செய்யும் நிரலை எழுதுக.

Chapter 11.indd 125 03/09/18 5:47 PM

N1KH1D

அணிகள் மற்றும் கட்டுருக்கள்

கற்றலின் நோக்கங்கள்

இப்பாடத்தை கற்றறிந்த பிறகு மாணவர்கள் தெரிந்து கொள்ளுதல்.

- கட்டுரு தரவினத்தை, அணிகளை பயன்படுத்தி அறிந்து கொள்ளுதல்.
- அணிகளின் வகைகளை அறிந்து கொள்ளுதல்.
- பல்வேறு வகையான அணிகளை கையாளுவதற்கு நிரல்களை எழுதுதல்.

12.1 முன்னுரை:

தரவுகளை சேமிக்க மாறிகள் உதவுகின்றன. C++ மொழியின் அடிப்படை தொகுதிகளாக மாறிகள் விளங்குகின்றன. நினைவகத்தில் ஒரு மதிப்பை எங்கு வேண்டுமானாலும்சேமிக்க மாறிபயன்படுகின்றது. சில சமயங்களில், நாம் ஒரே தரவினத்தை கொண்ட பல்வேறு மதிப்புகளை சேமிக்க அதே தரவினத்தை சார்ந்த பல்வேறு மாறிகளை பயன்படுத்த வேண்டும். நினைவகத்தில் இந்த மதிப்புகள் தோராயமான இடங்களில் சேமிக்கப்படும்.

எடுத்துக்காட்டாக, 100 மாணவர்களின் பதிவு எண்களை சேமிக்க, roll1 roll2 roll3 .... roll100 என்ற 100 மாறிகள் தேவைப்படும். 100 மாறிகளை அறிவித்தல், அதில் எல்லா பதிவு எண்களையும் சேமித்தல் என்பது மிக கடினமான செயலாகும். C++ மொழியில் பல மதிப்புகளை ஒரே மாறியில் சேமிக்க அணிகள் என்ற கருத்துரு உதவுகின்றது. பொதுவாக அணி என்றால் "ஒன்றிக்கு மேற்பட்ட" என்பதன் பொருளாகும். வேறுவிதமாக கூறினால், அணி என்பது “ஒன்றிக்கு மேற்பட்ட” ஒரே தரவின மதிப்புகளை பொதுவான ஒரே பெயரில் சேமிக்க ஒரு எளிய வழியாகும்.

C++ - ல் அணி என்பது ஓர் தருவிக்கப்பட்ட தரவினமாகும். “அணி என்பது ஒரே தரவினத்தைச் சார்ந்த மாறிகளின் திரட்டு ஆகும். அணியின் உறுப்புகளை ஒரு பொதுப்பெயரால் குறிப்பிடலாம்”. குறிப்பிட்ட எண்ணிக்கையிலான ஒரே தரவினத்தைச் சார்ந்த மதிப்புகளை நினைவகத்தில் தொடர்ச்சியான இடங்களில் சேமிக்கின்றன. எனவே, முழுஎண் தரவினத்தை சார்ந்த அணியானது முழுஎண்களின் வரிசையையும், குறியுரு தரவினத்தை சார்ந்த அணியானது குறியுரு வரிசையையும் கொண்டிருக்கும்.

அணியின் அளவானது அதன் பரிமாணமத்தால் குறிப்பிடப்படுகிறது.

12.2 அணிகளின் வகைகள்:

C++ மொழியில் பல்வேறு விதமான அணிகள் உள்ளனஅவை:

ஒரு பரிமாண அணிகள் (one dimensional arrays)

இரு பரிமாண அணிகள் (Two – dimensional arrays

பல பரிமான அணிகள் (Multi - dimensional arrays)

12.2.1 ஒரு பரிமாண அணி

இது ஒரு எளிய வடிவிலான அணி ஆகும். ஒரு பரிமாண அணியில் மதிப்புகள் ஒரே வரிசை

அல்லது ஒரே நெடுவரிசையில் சேமிக்கப்படும்.

அணியை அறிவித்தலுக்கான தொடரியல்:

<data types> <array-name> (<array - size>);

Data- type அணியின் தரவுவகையை குறிக்கும், அதாவது அணியிலுள்ள ஒவ்வொரு உறுப்பின் தரவுவகையாகும்.

Array-name அணியின் பெயரை குறிக்கும், அணியை அணுக இந்த பெயரே உதவும்.

Array-size அணியின் அளவை குறிப்பதற்கு பயன்படுகிறது. அணியின் அளவு எப்பொழுதும் [ ] என்ற அடைப்புக் குறிகளுக்குள் கொடுக்கப்படல் வேண்டும்.

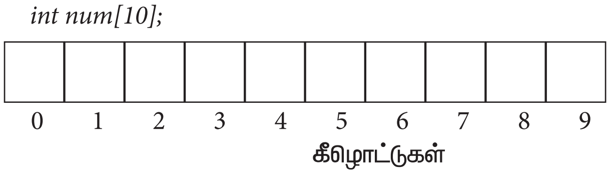
எடுத்துக்காட்டு:

int num [10] ;

மேலே அறிவிக்கப்பட்டுள்ள "num" என்ற அணியானது 10 முழு எண் மதிப்புகளை சேமிக்கும் அணியாகும். (முழு எண் தரவு வகையைச் சார்ந்த 10 பல்வேறு மதிப்புகளை நினைவகத்தில் சேமிக்கலாம்).

கீழே உள்ள படத்தில் உள்ளதுபோல, இந்த அறிவிப்புக்கு "num" என்ற பொதுப் பெயரில் நிரல்பெயர்ப்பி 10 நினைவக இடங்களை (பெட்டிகள்) ஒதுக்கும்.

int num[10];



ஒவ்வொரு உறுப்புக்கும் (நினைவக பெட்டி) தனித்தன்மையுடைய சுட்டு எண் 0 – விலிருந்து தொடங்கும், இந்த எண் "கீழொட்டு" என்றழைக்கப்படும். கீழொட்டு எப்பொழுதும் 0 என்ற எண்ணிலிருந்து தொடங்கும். மேலும் இது, குறியில்லா முழுஎண்ணாக இருக்க வேண்டும்.

அணியின் ஒவ்வொரு உறுப்பும் அதன் பொது பெயருடன் கூடிய சதுர அடைப்புக் குறிகளுக்குள் கொடுக்கப்பட்டிருக்கும் கீழொட்டு மதிப்பைகொண்டுகுறிக்கப்படும்.எடுத்துக்காட்டாக num [3] என்பது num என்ற அணியின் நான்காவது உறுப்பைச் சுட்டும்.

பல்வேறு தரவின வகையுடன் மேலும் பல அணிகளின் அறிவிப்புகள்:

char emp_name[25]; // emp-name என்ற அணி 25 உறுப்புகளை கொண்ட குறியுரு அணியாகும்.

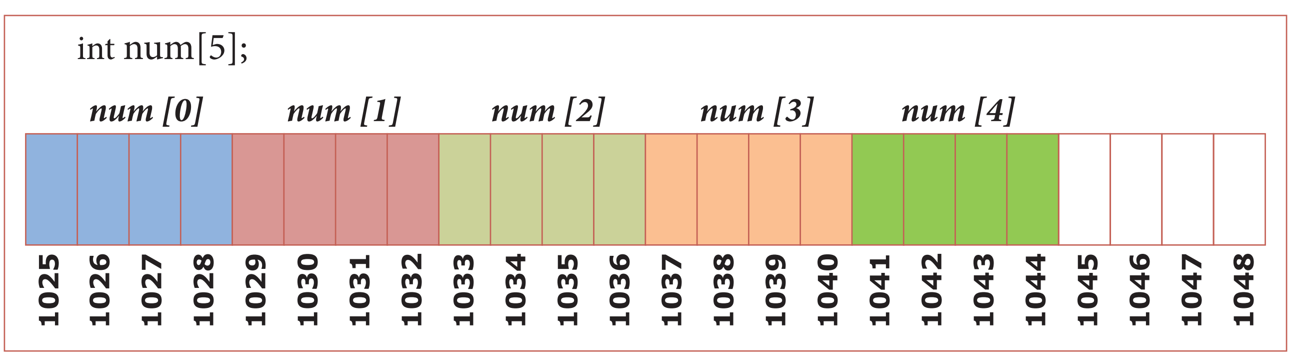
float salary[20];// salary என்ற அணி 20 தசமபுள்ளி உறுப்புகளை கொண்ட அணியாகும்.

int a [5] b[10], c[15]; // int தரவின வகையில் அமைந்த ஒன்றுக்கு மேற்பட்ட அணிகளின் அறிவிப்பு ஆகும்.

ஒரு பரிமாண அணியின் நினைவக ஒதுக்கீடு

ஒரு அணியின் நினைவக ஒதுக்கீடானது அதன் அளவு மற்றும் அதன் தரவுவகைக்கு நேரடியாக தொடர்புடையது.

கீழே கொடுக்கப்பட்டுள்ள படம் ஐந்து உறுப்புகளைக் கொண்ட அணியின் நினைவக ஒதுக்கீட்டை காட்டுகின்றது.



Chapter 12.indd 127 03/09/18 5:08 PM

மேலே கொடுக்கப்பட்டுள்ள படத்தில், num என்ற அணியானது 5 உறுப்புகளைக் கொண்ட முழு எண் அணி என்பது காட்டப்பட்டுள்ளது.Dev-C++நிரல்பெயர்ப்பியானது, ஒவ்வொரு int தரவுவகை மாறிக்கும் 4 பைட்டுகள் (bytes) நினைவகத்தை ஒதுக்கீடு செய்யும். 5 உறுப்புகளை கொண்ட இவ்வணிக்கு, ஒவ்வொரு உறுப்புக்கும் 4 பைட்டுகள் விதம் மொத்தமாக 20 பைட்டுகள் ஒதுக்கப்படும்.

தரவுவகை	நினைவகத்தில் byte –ல் ஒதுக்கீடு	
	Turbo C++	Dev C++
char	1	1
int	2	4
float	4	4
long	4	4
double	8	8

அணியின் நினைவக ஒதுக்கீட்டை கணக்கிட பயன்படும் வாய்பாடு

அணியின் தரவினத்துக்கு ஒதுக்கப்படும் பைட்டுகளின் எண்ணிக்கை × உறுப்புகளின் எண்ணிக்கை.

தொடக்க மதிப்பிருத்தல்

ஒரு அணியை அறிவிக்கும்பொழுதே அதற்கு உரிய தொடக்க மதிப்பை இருத்தலாம். அவ்வாறு அணிக்கு உரிய தொடக்க மதிப்பை இருத்தவில்லையெனில் அதன் உறுப்புகள் தேவையற்ற மதிப்புகளைக் (garbage values) கொண்டிருக்கும்.

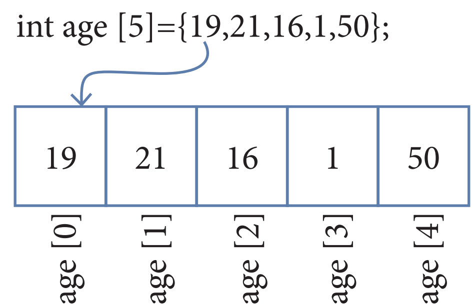
அணிக்கு தொடக்க மதிப்பு இருத்தலுக்கான தொடரியல்:

<data type> <array name> [size] = {value 1, value-2….. value –n};

எடுத்துக்காட்டு

int age [5] = { 19,21,16,1,50};

மேற்கண்ட எடுத்துக்காட்டில் 'age' என்ற அணியின் அளவானது 5 ஆகும். இதில் முதல் உறுப்பு age[0]ல் 19 என்ற எண் சேமிக்கப்பட்டிருக்கும், உறுப்பு age[1]ல் 21என்ற எண் சேமிக்கப்பட்டிருக்கும், இது போல மற்ற உறுப்புகளிலும் கொடுக்கப்பட்டுள்ள எண்கள் சேமிக்கப்பட்டிருக்கும். (பார்க்க படம் 12.1)



படம் 12.1

ஒரு அணியை அறிவிக்கும்போதே உறுப்புகளுக்கு தொடக்க மதிப்பு இருத்த வேண்டுமெனில், தொடக்க மதிப்புகள் கொக்கி வடிவ { } அடைப்புக்குறிக்குள் கொடுக்கப்படல் வேண்டும்.

அணியை அறிவிக்கும் போதே தொடக்க மதிப்பு இருத்தப்படுமாயின் அணியின் அளவை கொடுப்பது அவசியமில்லை.

எடுத்துக்காட்டு int age[ ] = {19, 21,16,1,50};

மேலே குறிப்பிட்டுள்ள எடுத்துக்காட்டில் தொடக்க மதிப்புடன் அணியை அறிவிக்கும்போது அணியின் அளவானது நேரடியாக கொடுக்கப்படவில்லை. ஆனால் நிரல்பெயர்ப்பி தானாகவே அணியில் இருத்தப்பட்ட மதிப்புகளின் எண்ணிக்கையைக் கொண்டு அணியின் அளவை கணக்கிடும்.

அணிக்கு தொடக்க மதிப்பிருத்தலுக்கான வேறுசில எடுத்துக்காட்டுகள்:

float x[5] = {5.6,5.7,5.8,5.9,6.1};

char vowel [6] = {'a', 'e', 'i' ,'o' ,'u', '\0'};

Chapter 12.indd 128 03/09/18 5:08 PM

நிரலை இயக்கும்போது அணிக்குறிய மதிப்புகளை உள்ளீடு செய்தல்:

நிரலின் இயக்க நேரத்தில் அணிக்கு ஒரு மதிப்பை மட்டுமே உள்ளீடு செய்ய முடியும். ஒன்றிற்க்கு மேற்பட்ட மதிப்புகளை உள்ளீடு செய்ய வேண்டுமெனில் ஏதேனும் ஒரு மடக்கை பயன்படுத்தலாம். for மடக்கானது அணியின் உறுப்புகளுக்கு மதிப்பை பெற சிறந்த மடக்கு வகையாகும்.

// நிரல் இயக்க நேரத்தில் மதிப்புகளை உள்ளீடாக பெறுதல்

```
#include <iostream>
using namespace std;
int main()
{
        int num[5];
        for(int i=0; i<5; i++)
        {
                cout<< "\n Enter value " << i+1 << "= ";
                cin>>num[i];
        }
}
```

மேலே கொடுக்கப்பட்டுள்ள நிரலில் int num [5]; என்பது 5 உறுப்புகளைக் கொண்ட ஒரு முழு எண் அணியாகும்.

```
for(int i=0; i<5; i++)
    {
        cout<< "\n Enter value " << i+1 << "= ";
        cin>>num[i];
    }
```

இந்த for மடக்கு 5 முறை இயங்குமாறு வடிவமைக்கப்பட்டுள்ளது. ஒவ்வொரு முறையும் மடக்கு இயங்கும் போதே “Enter value” என்ற அறிக்கையை cout கூற்று வெளியிடும். மேலும் cin மூலம் உள்ளீடு மதிப்பு பெறப்பட்டு num [i] என்ற அணியில் சேமிக்கப்படும்.

பின்வரும் அட்டவணையானது மேற்கண்ட நிரல் குறிமுறை செயல்படும் விதத்தை காட்டுகிறது.

Iteration	i <5	cout << "\n Enter value " << i+1 << "= ";	cin>>num [i];	நினைவகத்தில் சேமிக்கப்பட்ட உள்ளீடு மதிப்புகள்		i++ (i=i+1)
1	0 < 5 (T)	Enter value 1 =	num[0] = 5	num[0]	5	1
2	1 < 5 (T)	Enter value 2 =	num[1] = 10	num[1]	10	2
3	2 < 5 (T)	Enter value 3 =	num[2] = 15	num[2]	15	3
4	3 < 5 (T)	Enter value 4 =	num[3] = 20	num[3]	20	4
5	4 < 5 (T)	Enter value 4 =	num[4] = 25	num[4]	25	5
6	5 < 5 (F)	மடக்கிலிருந்து வெளியேறும்				

குறிப்பு : for மடக்கில் கட்டுப்பாட்டு மாறியை அறிவிக்கும்போது அதன் தொடக்க மதிப்பு 0 (பூஜ்ஜியம்) என்று இருத்தப்பட்டுள்ளது.

அணியின் உறுப்புகளின் மதிப்பை படிப்பதற்கோ அல்லது உறுப்புகளில் மதிப்பை சேமிப்பதற்கோ மடக்கினை அறிவிக்கும் போது மடக்கின் கட்டுப்பாட்டு மாறியின் தொடக்க மதிப்பானது 0 (பூஜ்ஜியம்) என அறிவிக்க வேண்டும். ஏனெனில் பெரும்பாலான சமயங்களில் கட்டுப்பாட்டு மாறியின் மதிப்பு அணியின் கீழொட்டை குறிக்க பயன்படுத்தப்படும்.

12.2.2 அணியின் உறுப்புகளை அணுகுதல்

அணியின் உறுப்புகளை நிரலில் நாம் எங்கு வேண்டுமானாலும் சாதாரண மாறிகளைப் போல பயன்படுத்தலாம். அணியின் உறுப்புகளை அணுக அணியின் பெயருடன் சதுர அடைப்புக் குறிகளுக்குள் கீழொட்டின் மதிப்பை கொடுக்க வேண்டும்.

எடுத்துக்காட்டு

cout << num [3];

இதில் num [3] என்ற கீழொட்டானது அணியின் 4வது உறுப்பை குறிக்கும். Num[3] என்ற உறுப்பின் மதிப்பை cout கூற்று வெளியிடும்.

அடைப்பு குறிக்குள் கொடுக்கப்படும் கீழொட்டு ஒரு மாறியாகவோ, ஒரு மாறிலியாகவோ அல்லது ஒரு கோவையாகவோ இருக்கலாம். கோவையாக இருந்தால் அதன் விடையானது ஒரு முழு எண் மதிப்பாகத்தான் இருக்க வேண்டும்.

```
// அணியின் உறுப்புகளை அணுகுதல்
#include <iostream>
using namespace std;
int main()
{
        int num[5] = {10, 20, 30, 40, 50};
        int t=2;
        cout<<num[2] <<endl;
        cout<<num[3+1] <<endl;
        cout<<num[t=t+1];
}
output:
        30
        50
        40
```

மேலே கொடுக்கப்பட்டுள்ள நிரலில், முதல் cout கூற்று num என்ற அணியின் 3வது உறுப்பின் மதிப்பை வெளியிடும் (கீழொட்டு சுட்டு 2) இரண்டாவது cout கூற்று அணியின் 5 வது உறுப்பின் மதிப்பை வெளியிடும் (கீழொட்டு மதிப்பு 3+1=4) மூன்றாவது cout கூற்று, 4-வது உறுப்பின் மதிப்பை வெளியிடும் (C++ மொழியானது சதுர அடைப்பு குறிக்குள் கொடுக்கப்பட்டுள்ள கோவையை t=2+1=3 என்று கணக்கிடும்.)

```
//கீழே கொடுக்கப்பட்டுள்ள நிரல் ஒரு அணியில் உறுப்புகளுக்கான மதிப்புகளை
எழுதுவது மற்றும் படிப்பதை விளக்குகின்றது.

#include <iostream>
using namespace std;
int main()
 {
          int age[4];//declaration of array
          cout<< "Enter the age of four persons:" <<endl;
          for(int i = 0; i < 4; i++)//loop to write array elements
                    cin>> age[i];
                    cout<<"The ages of four persons are:";
for(int j = 0; j< 4; j++)
                    cout<< age[j]<<endl;
}
```

இந்த நிரலில் இரண்டு மடக்குகள் உள்ளன. பின்வரும் அட்டவணையானது மேற்கண்ட நிரலின் செயல்பாட்டை காட்டுகிறது.

Iteration	i < 4	cin>> age [i];	Value received	age	i++ (i=i+1)
1	0 < 4 (T)	cin>> age[0];	18	age[0] = 18	1
2	1 < 4 (T)	cin>> age[1];	17	age[1] = 17	2
3	2 < 4 (T)	cin>> age[2];	21	age[2] = 21	3
4	3 < 4 (T)	cin>> age[3];	23	age[3] = 23	4
5	4 < 4 (F)	மடக்கிலிருந்து வெளிவரும்			

முதல் for மடக்கு மதிப்புகளை விசைப்பலகை மூலம் பெற்று நினைவகத்தில் சேமிக்கும். இரண்டாவது for மடக்கு சேமித்த மதிப்புகளை வெளியிடும்.

முதல் for மடக்கு

கீழே கொடுக்கப்பட்டுள்ள கூற்று மிகவும் முக்கியமானது ஏனெனில் மதிப்புகளை விசைப்பலகை வழியாக பெற்று நினைவகத்தில் சேமிக்கிறது.

```
for (int i = 0; i< 4; i++)
          cin>> age[i]>>endl;
```

மேலே கொடுக்கப்பட்டுள்ள கட்டளைகள் நிறைவேற்றப்பட்ட பின் உள்ளிடப்பட்ட மதிப்புகள் பின்வருமாறு நினைவகத்தில் சேமிக்கப்பட்டிருக்கும்.

18	17	21	23
0	1	2	3

கீழொட்டுகள்

இரண்டாவது for மடக்கு

```
for ( int j =0;j>4;j++)
          cout<<age[j]<<endl;
```

மேலே உள்ள கூற்றுகள் நினைவகத்திலிருந்து அணி உறுப்புகளின் மதிப்புகளை படித்து அவற்றை வெளியீடு செய்யும்.

Chapter 12.indd 131 03/09/18 5:08 PM

Iteration	j< 4	cout << age[j];	values read from memory	Output	j++
1	0 < 4 (T)	cout << age[0];	18	18	1
2	1 < 4 (T)	cout << age[1];	17	17	2
3	2 < 4 (T)	cout << age[2];	21	21	3
4	3 < 4 (T)	cout << age[3];	23	23	4
5	4 < 4 (F)	மடக்கிலிருந்து வெளியேறும்			

ஆகவே நிரலின் இறுதியான வெளியீடு இவ்வாறு இருக்கும்

Enter the age of four persons:

18

17

21

23

The ages of four persons are:

18

17

21

23

பயணித்தல் (traversal)

ஏதேனும் ஒரு செயல்பாட்டை செய்வதற்காக அணியில் உள்ள ஒவ்வொரு உறுப்பையும் ஒரு முறையாவது அணுகும் செயல்முறையே "பயணித்தல்" எனப்படும். அணியில் உள்ள அனைத்து உறுப்புகளின் மதிப்புகளை வெளியிடுவது பயணித்தலுக்கு ஒரு எடுத்துக்காட்டாகும்.

```
// அணியில் பயணித்தல்

#include <iostream>
using namespace std;
int main()
{
        int num[5];
        for (int i=0; i<5; i++)
        {
                cout<< "\n Enter value " << i+1 <<"= ";
                cin>>num[i];  // Reading from keyboard
```


Chapter 12.indd 132 03/09/18 5:08 PM

```
            num[i] = num[i] + 1; // Traversing the array elements sequentially
and adding 1 to each element
        }
        cout<< "\n After incrementing, the values in array num......" <<endl;
        for (int j=0; j<5; j++)
        {
            cout<<num[j]  <<endl;  //  Traversing  the  array  elements
sequentially and printing each one of them
        }
}
```

வெளியீடு:

```
        Enter value 1= 10
         Enter value 2= 20
         Enter value 3= 30
         Enter value 4= 40
         Enter value 5= 50
        After incrementing, the values in array num......
        11
        21
        31
        41
        51
```

10 மாணவர்களின் மதிப்பெண்களை படித்து அவற்றின் சராசரியை காண ஒரு நிரல்

```
#include <iostream>
using namespace std;
int main()
{
        int marks[10], sum=0;
        float avg;
        for(int i=0; i<10; i++)
         {
             cout<< "\n Enter Mark " << i+1 << "= ";
            cin>> marks[i];
            sum=sum+marks[i];
         }
        avg=sum/10,0;
        cout<< "\n The Total Marks: " << sum;
        cout<< "\n The Average Mark: " <<avg;
}
```

வெளியீடு:

```
Enter Mark 1= 41
Enter Mark 2= 98
Enter Mark 3= 65
Enter Mark 4= 75
Enter Mark 5= 35
Enter Mark 6= 82
Enter Mark 7= 64
Enter Mark 8= 5
Enter Mark 9= 58
Enter Mark 10= 68
The Total Marks: 591
The Average Mark: 59
```

C++ மொழியில் 10 மதிப்புகளை உள்ளீடாக பெற்று அதில் ஒற்றை எண்களின் எண்ணிக்கை (odd numbers) மற்றும் இரட்டை எண்களின் எண்ணிக்கையை (Even numbers) காண்பதற்கான நிரல்.

```
#include <iostream>
using namespace std;
int main()
{
        int num[10], even=0, odd=0;
        for (int i=0; i<10; i++)
         {
                cout<< "\n Enter Number " << i+1 <<"= ";
                cin>>num[i];
                if (num[i] % 2 == 0)
                 ++even;
                else
                 ++odd;
         }
         cout << "\n There are "<< even <<" Even Numbers";
         cout << "\n There are "<< odd <<" Odd Numbers";
}
```

வெளியீடு:

```
Enter Number 1= 78
Enter Number 2= 51
Enter Number 3= 32
Enter Number 4= 66
Enter Number 5= 41
Enter Number 6= 68
Enter Number 7= 27
Enter Number 8= 65
Enter Number 9= 28
Enter Number 10= 94
There are 6 Even Numbers
There are 4 Odd Numbers
```


Chapter 12.indd 134 03/09/18 5:08 PM

(HOT மேலே கொடுக்கப்பட்டுள்ள நிரலை if கூற்றிற்கு பதிலாக நிபந்தனைச் செயற்குறியைப் பயன்படுத்தி எழுதவும்)

ஏதேனும் 10 பொருள்களின் விலையை அணியின் மூலமாக அவற்றின் கூட்டுத்தொகை, மற்றும் சராசரி கணக்கிட்டு வெளியிடும் நிரல்.

```
#include <iostream>
using namespace std;
int main()
{
        float price[10], sum=0, avg=0, prod=1;
        for(int i=0; i<10; i++)
        {
                cout<< "\n Enter the price of item " << i+1 <<"= ";
                cin>> price[i];
                sum+=price[i];
                prod*=price[i];
        }
        avg=sum/10.0;
        cout<< "\n Sum of all prices: " << sum;
        cout<< "\n Product of all prices: " << prod;
        cout<< "\n Average of all prices: " <<avg;
}
```

ஒரு மாதத்தின் ஒவ்வொரு நாளின் விற்பனையை உள்ளீடாகப் பெற்று. அந்த மாதத்தின் மொத்த விற்பனை, கூட்டுத்தொகை கண்டறியும் நிரல்

```
#include <iostream>
using namespace std;
int main()
{
        int days;
        float sales[5], avgSales=0, totalSales=0;
        cout<< "\n Enter No. of days: ";
        cin>> days;
        for (int i=0; i<days; i++)
        {
                cout<< "\n Enter sales on day - " << i+1 <<": ";
                cin>> sales[i];
                totalSales+=sales[i];
        }
        avg=total sales/days;
        cout<< "\n Total Sales = " <<totalSales<<endl;
        cout<< "\n Average Sales = " <<avgSales;
}
```

கொடுக்கப்பட்ட எண்களின் தொகுப்பில் இருந்து ஒரு குறிப்பிட்ட மதிப்பை தேடும் செயல்முறை தேடல் என்றழைக்கப்படும். நேரியல் தேடல் (linear search) அல்லது தொடர் தேடல் ஆனது (Sequential search) தேடப்படும் மதிப்பை பட்டியலில் உள்ள உறுப்புகளின் மதிப்புடன் ஒப்பிட்டு அந்த மதிப்பை கண்டுபிடிக்கும்வரை அணியிலுள்ள அனைத்து உறுப்புகளுக்கும் பயணித்து ஒப்பிடும்.

நேரியல் தேடலுக்கான நிரல்

```
#include <iostream>
using namespace std;
int Search(int arr[], int size, int value)
{
        for (int i=0; i<size; i++)
{
        if (arr[i] == value)
        return i; // return index value
}
        return -1;
}
int main()
{
int num[10], val, id;
        for (int i=0; i<10; i++)
{
        cout<< "\n Enter value " << i+1 <<"= ";
        cin>>num[i];
}
        cout<< "\n Enter a value to be searched: ";
        cin>>val;
        id=Search(num,10,val);
        if(id==-1)
        cout<< "\n Given value is not found in the array..";
        else
        cout<< "\n The value isfound as the" << id+1 <<"th/ed element";
        return 0;
}
```

(Search) என்ற செயற் கூற்றை அழைத்து அதற்கு அணியின் பெயர், அளவு மற்றும் தேடவேண்டிய மதிப்பை அளபுருக்களாக அனுப்ப வேண்டும். தேடப்படும் மதிப்பு கண்டறியப்பட்டால் (அணியில் உள்ளதெனில்) அந்த மதிப்பின் கீழொட்டானது அழைத்த கூற்றுக்கு திரும்ப அனுப்பப்படும். இல்லையெனில் – 1 என்ற மதிப்பானது திருப்பி அனுப்பும்.

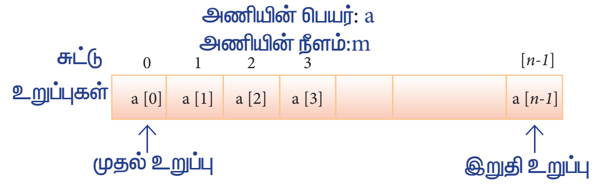
Chapter 12.indd 136 03/09/18 5:08 PM

சரங்கள்

குறியுருக்களின் வரிசையான சரம் என்கின்றோம், இதில் குறியுரு என்பது ஒரு எழுத்து எண் அல்லது குறியீடாக இருக்கலாம் ஒவ்வொரு குறியுருவும் நினைவகத்தில் ஒரு பைட் அளவு இடம் எடுத்துக்கொள்ளும். ஒவ்வொரு சரமும் அதன் முடிவை குறிக்கும் வெற்றுக் குறியுருவைக் (**null character-'\0', ASCII** கோடு **0**) கொண்டு முற்று பெற்றிருக்க வேண்டும். **C++** மொழியில் சரத்திற்கென தனி தரவு வகை கிடையாது. அதற்கு பதிலாக ஒரு சரத்தை ஒரு பரிமாண குறியுரு அணியாக அறிவிக்கலாம். குறியுரு அணியை அறிவிக்கும் போது, அது வெற்றிடக் குறியுருவை இறுதியில் கொண்டிருக்க வேண்டும். எனவே குறியுரு அணியின் அளவானது சரத்தின் நீளத்தை விட ஒரு குறியுரு அதிகமாக இருக்குமாறு அமைக்க வேண்டும்.

குறியுறு அணி (சரம்) உருவாக்குதல்

ஒரு அணியை உருவாக்கும் முன் அதனுடைய அளவை (நீளம்) அறிந்திருத்தல் வேண்டும். ஏனெனில் நினைவகமானது அணியின் அளவிற்கு தகுந்தவாறு ஒதுக்கப்படும். அணியானது உருவாக்கப்பட்ட பின்பு அதன் நீளத்தை இயக்க நேரத்தில் மாற்றமுடியாது. பார்க்க படம் 12.2-இல் காட்டுப்பட்டுள்ளது.



படம் 12.2 குறியுரு அணிக்கான நினைவக ஒதுக்கீடு

குறியுரு அணியை தெரிவிப்பதற்கான தொடரியல்:

char array- name [size];

மேலே கொடுக்கப்பட்டுள்ள தொடரியலில் அணியின் அளவு (Size) எப்பொழுதும் குறியில்லாத முழு எண் மதிப்பாக தான் இருக்கவேண்டும்.

எடுத்துக்காட்டு

char country [6];

மேற்கண்ட எடுத்துக்காட்டில் குறியுருக்களின் வரிசையை சேமிப்பதற்கு நினைவகத்தில் 6 பைட்டுகள் ஒதுக்கப்படும். சரத்தின் நீளம் 5 குறியுருக்களுக்கு மேல் இருக்க முடியாது. மேலும் வெற்று குறியுருக்காக ஒரு இடம் சரத்தின் இறுதியில் ஒதுக்கப்படும்.

//குறியுரு அணியை விளக்கும் நிரல்

```
#include <iostream>
using namespace std;
int main()
 {
char country[6];
          cout<< "Enter the name of the country: ";
          cin>>country;
          cout<<" The name of the country is "<<country;
}
```

வெளியீடு:

```
Enter country the name: INDIA
The country name is INDIA
```

தொடக்க மதிப்பிருத்தல்

குறியுறு அணியை அறிவிக்கும் போதே அதற்கு தொடக்கமதிப்பு இருத்தலாம். அதன் தொடரியல் கீழே கொடுக்கப்பட்டுள்ளது.

char arrayName[size]={ list of values separated by comma } ;

எடுத்துக்காட்டு

char country[6]="INDIA";

மேலே கொடுக்கப்பட்டுள்ள எடுத்துக்காட்டில் " INDIA" என்ற சரமானது 5 எழுத்துக்களை கொண்டுள்ளளது. இது country என்ற அணியில் தொடக்க மதிப்பாக இருத்தப்படுகிறது. உரையானது இரட்டை மேற்கோள் குறிகளுக்குள் கொடுக்கப்பட வேண்டும்.

Country[0]	Country[1]	Country[2]	Country[3]	Country[4]	Country[5]
I	N	D	I	A	'\0'
1000	1001	1002	1003	1004	1005

படம் 12.3 –ல் நினைவக ஒதுக்கீடு

மேற்கண்ட எடுத்துக்காட்டில் ஒவ்வொரு குறியுருவும் நினைவகத்தில் ஒரு பைட் எடுத்துக்கொள்ளும். சரத்தின் இறுதியில், நிரல்பெயர்ப்பி தானாகவே ஒரு வெற்று குறியுருவை அதன் இறுதியில் இணைத்துக்கொள்ளும்.

C++ மொழியில் குறியுரு அணிக்கு தொடக்க மதிப்பை இருத்துவதற்கான பல வழியில் கீழே கொடுக்கப்பட்டுள்ளது.

char country[6]={'I', 'N','D','I','A','\0'};

char country[]="INDIA";

char country[]={'I', 'N', 'D', 'I', 'A', '\0'};

அணியின் அளவு வெளிப்படையாக அறிவிக்கப்படவில்லையெனில் நிரல்பெயர்ப்பி தாமாகவே அதன் அளவை அணியின் உறுப்புகளின் எண்ணிக்கைக்கு ஏற்ப நிர்ணயிக்கும்.

சரத்திற்கு தொடக்க மதிப்பிருத்தும் போது அனைத்து உறுப்புகளுக்கும் தொடக்க மதிப்பு இருத்தப்படவில்லையெனில், மதிப்பிருத்தப்படாத மீதமுள்ள நினைவக இடங்களுக்கு வெற்று மதிப்பானது (NULL) இருத்தப்படும்.

எடுத்துக்காட்டு

char str[5]={'5','+','A'};

str[0]; ---> 5

str[1]; ---> +

str[2]; ---> A

str[3]; ---> NULL

str[4]; ---> NULL

சரங்களின் உறுப்புகளுக்கு தொடக்க மதிப்பிருத்தும்போது, அதன் அளவிற்கு அதிகமாக மதிப்புகளை இருத்தக்கூடாது.

எடுத்துக் காட்டு

char str[2]=]'5','+','A','B']; // தவறானது

மேலே கொடுக்கப்பட்டுள்ள எடுத்துக்காட்டில், நிரல்பெயர்ப்பி "Initialize string for array of chars in too long" என்ற பிழை அறிக்கையை வெளியிடும்.

Chapter 12.indd 138 03/09/18 5:08 PM

குறியுரு அணிகளுக்கு தொடக்க மதிப்பிருத்துவதற்காக பல்வேறு முறைகளை விளக்கும் நிரல்

```
#include <iostream>
#include<conio.h>
using namespace std;

int main()
{
        char arr1[6]="INDIA";
        char arr2[6]={'I','N','D','I','A','\0'};
        char arr3[]="TRICHY";
        char arr4[]={'T','R','I','C','H','Y','\0'};
        char arr5[8]="TRICHY";
        cout<<"arr1  is :" <<arr1<< " and its size is "<<sizeof(arr1)<<endl;
        cout<<"arr2  is :" <<arr2<< " and its size is "<<sizeof(arr2)<<endl;
        cout<<"arr3  is :" <<arr3<< " and its size is "<<sizeof(arr3)<<endl;
        cout<<"arr4  is :" <<arr4<< " and its size is "<<sizeof(arr4)<<endl;
        cout<<"The elements of arr5"<<endl;
        for(int i=0;i<8;i++)
        cout<<arr5[i]<<" ";
        getch();
        return 0;
}
```

வெளியீடு:

```
        arr1  is :INDIA and its size is 6
        arr2  is :INDIA and its size is 6
        arr3  is :TRICHY and its size is 7
        arr4  is :TRICHY and its size is 7
        The elements of arr5
        TRICHY
```

ஒரு வரி உரையை படிக்க

C++ மொழியில் cin.get( ) என்ற செயற்கூறானது வெற்று இடைவெளிகளுடன் கூடிய ஒரு முழு நீள வரியை உள்ளீடாக பெற பயன்படுகிறது. இந்த செயற்கூறு இரண்டு செயலுருபுகளை ஏற்கும். முதல் செயலுருபு சரத்தின் பெயரையும் இரண்டாவது செயலுருபு அணியின் அதிகப்படியான அளவையும் குறிக்கும்.

get( )செயற்கூறுவை பயன்படுத்தி ஒரு வரி உரையை உள்ளிடுவதற்கான நிரல்

```
// str10.cpp
// To read a line of text
#include <iostream>
using namespace std;
int main()
{
charstr[100];
```

```
cout<< "Enter a string: ";
cin.get(str, 100);
cout<< "You entered: " <<str<<endl;
return 0;
}
```

வெளியீடு:

```
Enter a string: I am a student
You entered: I am a student
```

மேலே கொடுக்கப்பட்டுள்ள நிரலில் str என்பது சரத்தின் பெயர் மேலும் இந்த குறியுரு அணியின் அதிகபட்ச அளவு 100 என்று குறிப்பிடப்பட்டுள்ளது.

C++ மொழியில் getline () என்ற செயற்கூறும் ஒரு முழு வரியிலான இடைவெளிகளுடன் கூடிய உரையை உள்ளீட்டு தாரையின் மூலம் உள்ளீடு செய்யப் பயன்படுகிறது. இது குறியுருக்கான நகர்த்தி திரும்பலை (newline charcter) அல்லது வரம்புக்குறியை (delimiter) பயனர் உள்ளீடு செய்யும் வரை உள்ளீட்டை ஏற்கும். <string> என்ற தலைப்பு கோப்பில் இந்த செயற்கூறுக்கு உரிய அறிவிப்புகள் உள்ளது.

கொடுக்கப்படும் சொல்லானது palindrome சொல்லா என சோதிக்க C++ நிரல் எழுதுதல்.

string_array.cpp | Char_arr1.cpp

```
// palindrome using character array
#include<iostream>

using namespace std;

int main( )
{
        int i,j,len,flag =1;
        char a [20];

        count<<"Enter a String:";
        cin>>a;

        for(len=0;a[len]!='\0';++len);

        for(!=0,j=len-1;i<len/2;++i,--j)
        {
                if(a[j]!=a[i])
                        flag=0;
        }

        if(flag==1)
                count<<"\n The String is palindrom";
        else
                cout<<"\nthe string is not palindrome";

        return 0;
}
```

வெளியீடு

```
Enter a string : madam
The String is palindrome
```


Chapter 12.indd 140 03/09/18 5:08 PM

இரு பரிமாண அணி

இரு பரிமாண அணி என்பது ஒரே தரவினத்தைசார்ந்தஉறுப்புகளின்தொகுதியாகும். மேலும் இவை குறிப்பிட்ட எண்ணிக்கையிலான வரிசை மற்றும் நெடுவரிசையில் சேமிக்கப்படும். எடுத்துக்காட்டாக mxn அணிக்கோவையில் m என்பது வரிசையின் எண்ணிக்கையும், n என்பது நெடுவரிசையின் எண்ணிக்கையும் ஆகும் இது படம் 12.4- ல் காட்டப்பட்டுள்ளது. int arr [3] [3];

2D அணியின் கருத்து வடிவ நினைவக குறிப்பீடு

நெடுவரிசை கீழொட்டு →

வரிசை கீழொட்டு ↓

arr[0] [0]	arr[0] [1]	arr[0] [3]
arr[1] [0]	arr[1] [1]	arr[1] [2]
arr[2] [0]	arr[2] [1]	arr[2] [2]

arr என்ற அணி 3 வரிசைகள் மற்றும் 3 நெடுவரிசைகளுடன் கருத்து ரீதியான அணி வடிவத்தில் பார்க்கப்படுகிறது. கீழொட்டு 0 arr [0][0] என்பது முதல் உறுப்பை குறிக்கின்றது.

படம் 12.4

இரு பரிமாண அணியின் அறிவிப்பு:

இரு பரிமாண அணியை அறிவிப்பதற்கான தொடரியல்

data-type array- name [row size] [col-size];

மேலே குறிப்பிடப்பட்டுள்ள அறிவிப்பில் data type என்பது C++ மொழியில் உள்ள முறைப்படியிலான ஏதேனும் ஒரு தரவினத்தை குறிக்கும். Array-name என்பது 2D அணியின் பெயர், row size என்பது வரிசைகளின் எண்ணிக்கையையும், col-size என்பது நெடுவரிசைகளின் எண்ணிக்கையையும் குறிக்கும்.

எடுத்துக்காட்டு

int A [3] [4];

மேலே குறிப்பிடப்பட்டுள்ள எடுத்துக்காட்டில் A என்ற இரு பரிமாண அணியானது 3 வரிசைகள் மற்றும் 4 நெடுவரிசைகளை கொண்டுள்ள அணியாகும். இது அதிகபட்சமாக 12 உறுப்புகளை ஏற்கும்.

குறிப்பு: அணியின் அளவு எப்பொழுதுமே குறியில்லா முழுஎண் மதிப்பாக இருக்கவேண்டும். மேலும் இது 0 என்ற எண்ணின் மதிப்பை விட கூடுதலாக இருக்கவேண்டும். பயனர் அணியின் அளவை தெரிவிக்கவில்லையெனில், நிரல்பெயர்ப்பி பிழை அறிக்கையை வெளியிடும்.

இரு பரிமாண அணியை அறிவிக்க வேறுபல எடுத்துக்காட்டுகள்:

int A[3] [3];

float x [2] [3];

char name [5] [20];

12.3.2 இரு பரிமாண அணிக்கு தொடக்க மதிப்பிறுத்தல்

இரு பரிமாண அணியை அறிவிக்கும் போதே அதற்கு ஒன்றுக்கு மேற்பட்ட வழிகளில் தொடக்க மதிப்பிருத்தலாம்.

எடுத்துக்காட்டு

```
int matrix [4] [3] = {10,20,30,40,50,60,70,
80,90,100,110,120};
{10,20,30}, // வரிசை எண் 0 ல் தொடக்க மதிப்பிருத்தும்
{10,20,30}, // வரிசை எண் 1 ல் தொடக்க மதிப்பிருத்தும்
{70,80,90} // வரிசை எண் 2- ல் தொடக்க மதிப்பிருத்தும்
{100,110,120}; // வரிசை எண் 3- ல் தொடக்க மதிப்பிருத்தும்
};
```

இரு பரிமாண அணியை தொடக்க மதிப்புடன் அறிவிக்கும் போது அணியின் வரிசையின் அளவை குறிப்பிட வேண்டிய அவசியமில்லை, ஆனால் நெடுவரிசையின் அளவை கண்டிப்பாக குறிப்பிடவேண்டும்.

Chapter 12.indd 141 03/09/18 5:08 PM

எடுத்துக்காட்டாக,

```
int matrix[][3]={
{10,20,30},// row 0
{40,50,60},// row 1
{70,80,90},// row 2
{100,110,120}// row 3
};
```

12.3.3 இரு பரிமாண அணியை அணுகுதல்

இருபரிமாண அணியில் ஒரு உறுப்பை அணுகுவதற்கு இரண்டு சுட்டு மதிப்புகளை பயன்படுத்த வேண்டும். இதில் முதல் சுட்டு வரிசையின் மதிப்பையும் இரண்டாவது சுட்டு நெடுவரிசையின் மதிப்பையும் குறிக்கும்.

matrix [0] [0] =10; // முதல் வரிசையில் உள்ள முதல் உறுப்பிற்கு 10 என்ற மதிப்பிருத்தும்.

matrix [0] [1] =20; // முதல் வரிசையில் உள்ள இரண்டாவது உறுப்பிற்கு 20 என்ற மதிப்பிருத்தும்

matrix [0] [2] =60; // இரண்டாம் வரிசையில் உள்ள மூன்றாம் உறுப்பிற்கு 60 என்ற மதிப்பிருத்தும்

matrix [0] [3] =100; // முதல் வரிசையில் உள்ள நான்காம் உறுப்பிற்கு 100 என்ற மதிப்பிருத்தும்.

இரு அணிக்கோவைகளில் கூட்டுத்தொகையை கணக்கிட தேவையான நிரல்

இரண்டு அணிகளை கூட்டுவதற்கான சி ++ நிரல் எழுதுதல்

```
#include<iostream>
#include<conio.h>
using namespace std;
int main()
{
int row, col, m1[10][10], m2[10][10], sum[10][10];
cout<<"Enter the number of rows : ";
cin>>row;
cout<<"Enter the number of columns : ";
cin>>col;
cout<< "Enter the elements of first 1st matrix: "<<endl;
for (int i = 0;i<row;i++ ) {
for (int j = 0;j <col;j++ ) {
cin>>m1[i][j];
}
}
cout<< "Enter the elements of second 2nd matrix: "<<endl;
for (int i = 0;i<row;i++ ) {
for (int j = 0;j <col;j++ ) {
cin>>m1[i][j];
}
}
```

```
cout<< "Enter the elements of second 2nd matrix: "<<endl;
for (int i = 0;i<row;i++ ) {
for (int j = 0;j<col;j++ ) {
cin>>m2[i][j];
}
}
cout<<"Output: "<<endl;
for (int i = 0;i<row;i++ ) {
for (int j = 0;j<col;j++ )
{
sum[i][j]=m1[i][j]+m2[i][j];
cout<<sum[i][j]<<" ";
}
cout<<endl<<endl;
}
getch();
return 0;
}
```

```
Enter the number of rows :   2
Enter the number of column : 2
Enter the elements of first 1st matrix:
1
1
1
1
Enter the elements of first 1st matrix:
1
1
1
1
Output:
2   2
2   2
```

12.3.4 இரு பரிமாண அணியின் நினைவக ஒதுக்கீடு

பொதுவாக இரு பரிமாண அணியை ஒரு அணிக்கோவையாக காட்டலாம். இரு பரிமாண அணியின் கருத்து வடிவ படம் கீழே கொடுக்கப்பட்டுள்ளது.

Int A[4] [3]

Chapter 12.indd 143 03/09/18 5:08 PM

A[0][0]	A[0][1]	A[0][2]
A[1][0]	A[1][1]	A[1][2]
A[2][0]	A[2][1]	A[2][2]
A[3][0]	A[3][1]	A[3][2]

மேலே கொடுக்கப்பட்டுள்ள எடுத்துக்காட்டில் A என்ற இரு பரிமாண அணியானது 4 வரிசைகள் மற்றும் 3 நெடு வரிசைகளை கொண்டுள்ளது. ஒரு பரிமாண அணியை போல, இரு பரிமாண அணி உறுப்புகளும் நினைவகத்தில் தொடர்ச்சியாகவே சேமிக்கப்பட்டிருக்கும். இரு வகைகளில் இரு பரிமாண அணியின் நினைவக ஒதுக்கீட்டை குறிப்பிடலாம். அவை:

வரிசை முதன்மை கொண்டு அணுகுதல் (Row Major order)

நெடுவரிசை முதன்மை கொண்டு அணுகுதல் (Column Major Order)

எடுத்துக்காட்டு

Int A [4] [3] ={ {8,6,5}, {2.1.9},{3,6,4},{4,3,2},

வரிசையை முதன்மையாக கொண்டு அணுகுதல்

வரிசையை முதன்மையாக கொண்டு அணுகுதல் முறையில் அனைத்து உறுப்புகளும் ஒவ்வொரு வரிசையாக நினைவக இடத்தில் தொடர்ச்சியாக சேமிக்கப்பட்டிருக்கும். அதாவது முதல் வரிசையில் எல்லா உறுப்புகளும் முதலில் சேமிக்கப்பட்ட பின்பு அடுத்து இரண்டாவது வரிசையின் எல்லா உறுப்புகளும் சேமிக்கப்படும் அதேபோன்று அனைத்து வரிசையின் உறுப்புகளும் சேமிக்கப்படும். வரிசை முதன்மையாக கொண்டு அணுகுதல் முறையில் நினைவக ஒதுக்கீட்டு முறையை கீழே காணலாம்.

8	6	5	2	1	9	3	6	4	4	3	2
1000	1004	1008	1012	1016	1020	1024	1028	1032	1036	1040	1044

வரிசை 0 வரிசை 1 வரிசை 2 வரிசை 3

நெடுவரிசையை முதன்மையாக கொண்டு அணுகுதல்

நெடுவரிசையை முதன்மையாக கொண்டு அணுகுதல் முறையில் அனைத்து உறுப்புகளும் ஒவ்வொரு நெடுவரிசையாக நினைவக இடத்தில் தொடர்ச்சியாக சேமிக்கப்பட்டிருக்கும். அதாவது முதல் நெடுவரிசையின் எல்லா உறுப்புகளும் முதலில் சேமிக்கப்பட்ட பின்பு அடுத்து இரண்டாவது நெடுவரிசையின் எல்லா உறுப்புகளும் சேமிக்கப்பட்டும் அதேபோன்று அனைத்து நெடுவரிசையின் உறுப்புகளும் சேமிக்கப்படும். நெடுவரிசையை முதன்மையாக கொண்டு அணுகுதல் முறையில் நினைவக ஒதுக்கீட்டு முறையை கீழே காணலாம்.

8	2	3	4	6	1	6	3	5	9	4	2
1000	1004	1008	1012	1016	1020	1024	1028	1032	1036	1040	1044

Col 0 Col 1 Col 2

Chapter 12.indd 144 03/09/18 5:08 PM

12.4 சரங்களின் அணி

சரங்களின் அணி என்பது ஒரு இரு பரிமாண குறியுரு அணியாகும். அணி வரையறுப்பில் உள்ள முதல் சுட்டெண் (வரிசை) சரங்களின் எண்ணிக்கையைக் குறிக்கும். இரண்டாவது சுட்டெண் (நெடுவரிசை) சரங்களின் உச்ச அளவு நீளத்தைக் குறிக்கும். பொதுவாக, சரங்களின் அணியை அறிவிக்கும் போதே ஒவ்வொரு சரத்தின் இறுதியிலும் வெற்றுக் குறியுருவை இணைப்பதற்கு இடமளிக்கும் வகையில் அறிவிக்கப்படல் வேண்டும். எடுத்துக்காட்டாக கீழே உள்ள இரு பரிமாண அணியை அறிவித்தலை காண்போம்.

```
char Name[6][10];
```

மேற்கண்ட அறிவிப்புக் கூற்றில் இரு பரிமாண அணியில் இரண்டு சுட்டுகள் உள்ளது. அவை வரிசையின் அளவையும் மற்றும் நெடுவரிசையின் அளவையும் குறிக்கின்றது. அதாவது 6 என்பது வரிசைகளின் எண்ணிக்கையும் மற்றும் 10 என்பது நெடுவரிசைகளின் எண்ணிக்கையும் குறிக்கும்.

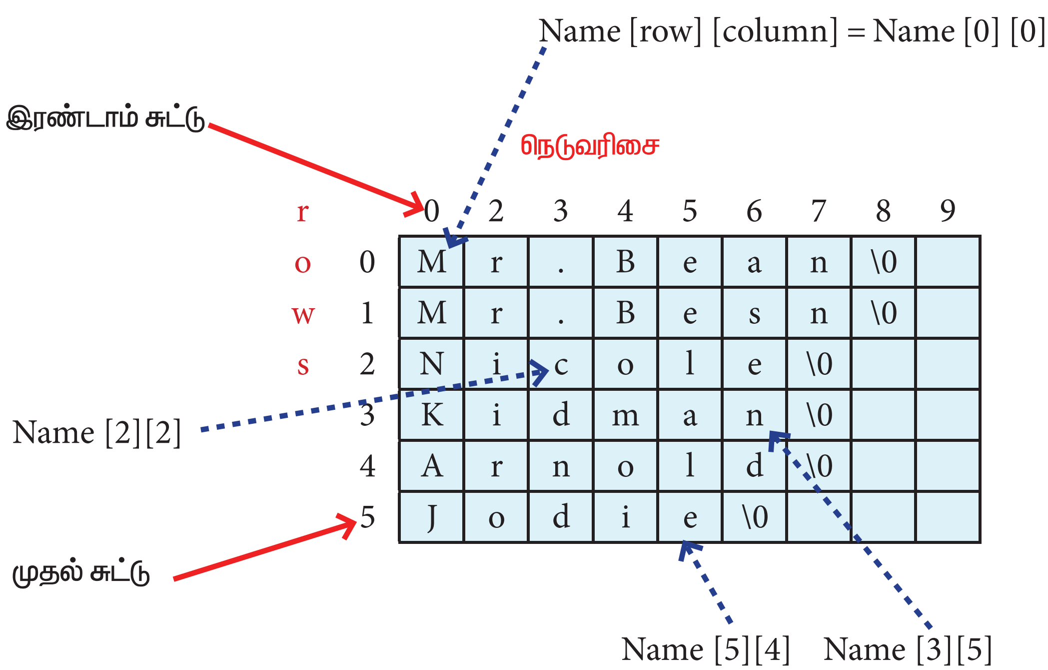
12.4.1 தொடக்கமதிப்பிருத்துதல்

எடுத்துக்காட்டாக,

```
char Name[6][10] = {"Mr. Bean", "Mr.Bush", "Nicole", "Kidman", "Arnold", "Jodie"};
```

மேற்கண்ட எடுத்துக்காட்டில் இரு பரிமாண அணி 6 சரங்களுடன் தொடக்க மதிப்பிருத்தப்பட்டுள்ளது. இதில் ஒவ்வொரு சரத்தின் உச்ச குறியுறு அளவின் மதிப்பு 9 ஆகும்.

இரு பரிமாண அணியின் நினைவக ஒதுக்கீடு மற்றும் கொடுக்கப்பட்ட அனைத்து சரங்களும் தொடர்ச்சியான இடங்களில் சேமிக்கப்பட்டிருப்பதை கீழே காணலாம்.



இரு பரிமாண குறியுரு அணியை பயன்படுத்தி சரங்களின் அணியை சேமித்தல்

```
#include<iostream>
using namespace std;
int main( )
{
        // initialize 2d array
        char colour [4][10]={"Blue","Red","Orange",
                             "yellow"};

        // printing strings stored in 2d array
        for (int i=0;  i <4; i++)
        cout  <<  colour [i] << "\n";
}
```

Output:

```
        Blue
        Red
        Orange
        Yellow
```

12.5 அணிகளை செயற்கூறுகளுக்கு அளபுருவாக அனுப்புதல்

C++ மொழியில் அணிகளை செயற்கூற்றிற்கு அளபுருவாக அனுப்பலாம்.

C++ மொழியில் அணியை செயற்கூற்றிக்கு அனுப்பும் போது, அணியின் பெயரை அளபுருவாக குறிப்பிட வேண்டும்.

செயற்கூற்றுக்கு இரு பரிமாண அணியை அனுப்ப

5 மாணவர்களின் மதிப்பெண்களை கொண்ட ஒரு பரிமாண அணியை செயற்கூறாக அனுப்பி மதிப்பெண்களை வெளியிடும் நிரல்.

[*] linear_search.cpp | fun_with_arr1.cpp

```
// palindrome using character array
#include<iostream>
using namespace std;

void display (int marks[5];

int main()
{
        int marks[5] = {88,76,90,61,69};
        display(marks);
        return 0;
}

void display(int m[5])
{
        cout <<"Displaying marks: "<< end1;

        for (int i = 0; i < 5 ; ++i)
        {
                cout << "Student " << i + 1 <<": "<< m [i] << end 1;
        }
}
```


Chapter 12.indd 146 03/09/18 5:08 PM

```
Displaying marks:
Student 1: 88
Student 2: 76
Student 3: 90
Student 4: 61
Student 5: 69
```

மேலே கொடுக்கப்பட்டுள்ள எடுத்துக்காட்டில், marks என்ற அணியை display என்ற செயற்கூற்றிற்கு அளபுருவாக அனுப்பி வெளியீடு பெறப்பட்டுள்ளது.

12.6 இரு பரிமாண அணியை செயற்கூறிற்கு அனுப்புதல்

இருபரிமாண அணியில் உள்ள மதிப்புகளை வெளியிடும் நிரல்

[*] linear_search.cpp | fun_with_arr1.cpp | fun_with_arr2.cpp

```
// function with two-dimensional array
#include <iostream>
using namespace std;

void display(int n[3][2]);

int main()
{
    int num[3][2] = {
        {3, 4},
        {9, 5},
        {7, 1}

    };
    display(num);

    return 0;
}

void display(int n[3][2])
{

    cout << "Displaying Values: " << endl;
    for(int i = 0;  i < 3; ++i)
    {
        for(int j = 0; j < 2; ++j)
        {
            cout << n[i][j] << " ";
        }
        cout<<endl<<endl;
```


Chapter 12.indd 147 03/09/18 5:08 PM

```
Displaying values:
3 4
9 5
7 1
```

மேலே கொடுக்கப்பட்டுள்ள நிரலில் num என்ற இரு பரிமாண அணியானது display() என்ற செயற்கூற்றிற்கு அளபுருவாக அனுப்பி வெளியீடு பெறப்பட்டுள்ளது.

குறியுரு அணியை அளபுருவாக கொண்ட செயற்கூறு

```
#include <iostream>
using namespace std;
int main()
{
        char str[100];
void display(char s[]);
        cout<< "Enter a string: ";
        getline(cin, str);
display(str);
        return 0;
}

void display(char s[])
{
cout<< "You entered char array: " << s <<endl;
}
```

output

```
        Enter a string: welcome to C++ programming
        You entered char array: welcome to C++ programming
```

ஆய்வு அறிக்கை

1. 10 மாணவர்களின் மதிப்பெண்களை உள்ளீடாக பெற்று அவற்றின் சராசரி அவற்றுள் அதிகமான மதிப்பெண் மற்றும் குறைவான மதிப்பெண்களை காண ஒரு நிரலை எழுதுக.

2. இந்தியாவில் உள்ள நான்கு முக்கிய நகரங்களின் மழைநீர் அளவை உள்ளீடாகப் பெற்று அவற்றுள் அதிகமான மழையின் அளவு மற்றும் குறைந்த மழை அளவு கொண்ட நகரங்களை காண ஒரு நிரலை எழுதுக.

Chapter 12.indd 148 03/09/18 5:08 PM

3. உங்கள் அருகாமையில் உள்ள கடைகளை ஆய்வு செய்து அதில் உங்களுக்கு விருப்பமான ஒரே ஒரு பொருளின் விலையை உள்ளீடாக பெற்று, மேலும் எந்த கடையில் அந்த பொருளை குறைவான விலையில் வாங்க இயலும் என்ற ஆய்வறிக்கையை வெளியிட நிரல் எழுதுக.

மதிப்பீடு

N1PIPA

பகுதி அ

I சரியான விடை யை தேர்வு செய்யவும் :

1. இவற்றுள் எது ஒரே தரவினத்தைச் சேர்ந்த மாறிகளின் திரட்டு மற்றும் அனைத்து உறுப்புகளையும் ஒரே பொதுப் பெயரால் குறிப்பிட இயலும்?

   அ) int ஆ) float இ) Array ஈ) class

2. அணியின் கீழொட்டு எப்பொழுதும் எந்த எண்ணுடன் தொடங்கும் ?

   அ)-1 ஆ) 0 இ 2 ஈ) 3

3. Int age[]={6,90,20,18,2}; இந்த அணியில் எத்தனை உறுப்புகள் உள்ளன?

   அ) 2 ஆ) 5 இ) 6 ஈ) 4

4. Cin>>n[3]; இந்த கூற்று எந்த உறுப்பில் மதிப்பை உள்ளீடும்?

   அ) 2 ஆ) 3 இ) 4 ஈ) 5

5. சரங்கள் தானமைவாக இவற்றுள் எந்த குறியுருவுடன் முடிவடையும்?

   அ) \o ஆ) \t இ) \n ஈ) \b

பகுதி– ஆ

II குறுவினாக்கள் (2 மதிப்பெண்கள்)

1. அணியில் பயணித்தல் என்றால் என்ன?
2. சரங்கள் என்றால் என்ன?
3. இரு பரிமாண அணியை அறிவிக்கும் தொடரியலை எழுதுக.

பகுதி– இ

III சிறுவினாக்கள் (3 மதிப்பெண்கள்)

1. அணி என்றால் என்ன ? அதன் வகைகளை எழுதுக.
2. சரங்களின் அணியைப் பற்றி சிறுகுறிப்பு எழுதுக.
3. C++ மொழியில் உனது பெயரை உள்ளீடாக பெற்று வெளியீட ஒரு நிரலை எழுதுக?

பகுதி – ஈ

IV பெருவினாக்கள் (5 மதிப்பெண்கள்)

1. இரண்டு அணிக்கோவைகளில் உள்ள மதிப்புகளின் வித்தியாசம் கண்டறிய நிரலை எழுதுக.
2. இரு பரிமாண அணியை செயற்கூற்றிற்கு அனுப்பும் முறையை ஒரு எடுத்துக்காட்டுடன் விளக்குக.

கற்றலின் நோக்கங்கள்

இப்பாடத்தை கற்றறிந்த பிறகு மாணவர்கள் தெரிந்து கொள்ளுதல்.

- பயனர் வரையறுக்கம் தரவின வகைகளின் நோக்கம் பற்றி புரிந்து கொள்ளுதல்.
- கட்டுருக்களைப் பயன்படுத்தி C++ நிரல்களை உருவாக்கும் முறையை பற்றி அறிதல்.
- கட்டுரு தரவின வகையை பயன்படுத்தி உருவாக்கப்படும் நிரல்களில் பிழைகள் கண்டறிந்து செயல்படுத்துதல் பற்றி அறிதல்.

12.7 கட்டுருக்கள் அறிமுகம்

வெவ்வேறு வகையான தரவு இனங்களை கொண்ட பயனர் வரையறுக்கும் தரவினம் கட்டுரு எனப்படும். இது பல்வேறு வகையான தரவு இனங்களுடன் கூடிய மாறிகளை ஒரே தொகுதிக்குள் ஒன்றிணைத்துள்ளது.

12.7.1. கட்டுருக்களின் நோக்கம்

ஒரே மாதிரியான தரவு வகைகளை சார்ந்த ஒன்றுக்கு மேற்பட்ட மாறிகள் தேவைப்படும் போது அவற்ற குறிக்க அணிகள் பயன்படுத்தப்படுகிறது. வெவ்வேறு தரவு வகை சார்ந்த உறுப்புகளாக இருந்தால் அணிகளை பயன்படுத்த இயலாது. ஒன்றுக்கு மேற்பட்ட மாறிகளை பயன்படுத்தினால் அவை நினைவகத்தில் சேமிக்கப்படும் போது ஒன்றுக்கொன்று அருகாமையான இடங்களில் இருக்காது. இதனால் தேடும் போது தேடலுக்கான நேரம் அதிகரிக்கிறது. வெவ்வேறு தரவு வகையை சார்ந்த தரவு உறுப்புகளை ஒரே தொகுதியின் அறிவித்து, அவைகளுக்கு நினைவகத்தில் அருகருகே இடம் ஒதுக்க கட்டுரு உதவுகிறது.

12.7.2 கட்டுருக்களை அறிவித்தல் மற்றும் வரையறுத்தல்

struct என்பது C++ மொழியின் சிறப்பு சொல். இது ஒரு கட்டுருவை அறிவிக்க பயன்படுகிறது. ஒரு கட்டுருவை உருவாக்குவதற்கான தொடரியல் கீழே கொடுக்கப்படடுள்ளது.

```
struct structure_name {
        data type member_name1;
        data type member_name2;
} reference_name;
```

> கட்டுரு வரையறுப்புடன் அறிவிக்கப்பட்டுள்ள பொருள்கள் உலகலாவிய (gopal objects) பொருள்கள் என்று அழைக்கப்படும்.

கட்டுரு தரவினத்தை அறிவிக்கும் போதே அதில் நேரடியாக மாறிகளை உருவாக்குதல் என்பது விருப்ப தேர்வு ஆகும்.

எ-கா:

> கட்டுருவில் உள்ள அனைத்து மாறிகளும் (members) தானமைவாக (public) அனுகியல்பை சார்ந்துள்ளன.

struct Student

Chapter 12.indd 150 03/09/18 5:08 PM

```
{
        long rollno;
        int age;
        float weight;
} ;
```

மேற்கண்ட கட்டுரு அறிவிப்பில் rollno, age, weight மூன்று மாறிகள் அறிவிக்கப்பட்டுள்ளன. கட்டுருக்களில் அறிவிக்கப்படும் இந்த மாறிகள் உறுப்பினர்கள் (Members) என்று அழைக்கப்படுகின்றது. student என்ற கட்டுரு தரவினத்தை பயன்படுத்துவதற்கு அதில் மாறிகளை அறிவித்தல் வேண்டும். student என்ற கட்டுரு தரவினத்தில் மாறிகளை அறிவித்தலும் அதற்கான நினைவக ஒதுக்கீடும் படம் 12.5 ல் காட்டப்பட்டுள்ளது.

Rollno	Age	weight
4 பைட்டுக்கள்	2 பைட்டுக்கள்	4 பைட்டுக்கள்

படம் 12.5 நினைவக ஒதுக்கீடு

struct student balu; // மாணவர் என்ற கட்டுரு தரவினத்தில் மாறியை அறிவித்தல்

student என்ற தரவினத்தில் balu என்ற மாறியானது அறிவிக்கப்பட்டுள்ளது. சாதாரண மாறிகளை கட்டுரு தரவின மாறிகளும் தங்களுக்குரிய நினைவக ஒதுக்கீட்டை தாங்களே மேற்கொள்கின்றன. இதில் ஒன்றுக்கும் மேற்பட்ட மாறிகளை ஒரே கட்டுரு வகையில் வரையறுக்க முடியும்.

உதராணமாக Balu மற்றும் Frank - என்ற இரண்டு மாறிகளை கட்டுரு தரவு வகையின் பொருள்களாக கீழ் வருமாறு அறிவிக்க முடியும்.

```
struct Student
{
long rollno;
int age;
float weight;
}balu, frank;
```

12.7.3 கட்டுரு உறுப்புகளை அணுகுதல்

student என்ற கட்டுரு தரவினத்தில் இரண்டு இரண்டு பொருள்களை அறிவித்தவுடன் அவறிற்கான தரவு உறுப்புகளை நேரடியாக அணுகலாம். பொருளானது தரவு உறுப்புகளை அணுகுவதற்கு பொருளின் பெயர் மற்றும் இரண்டு பொருள்களை அறிவித்தவுடன் அதன் உறுப்புகள் அவற்றை நேரடியாக அணுக முடியும். அதற்கான தொடரியலில் பொருளின் பெயர் மற்றும் உறுப்பினரின் பெயருக்கு இடையில் ஒரு புள்ளி (.) பயன்படுத்தப்படுகிறது.எடுத்துக்காட்டாக மாணவர் கட்டுரு அதன் உறுப்புகளை பின்வருமாறு அணுகலாம்:

```
balu.rollno
balu.age
balu.weight
frank.rollno
frank.age
frank.weight
```

பெயரற்ற கட்டுருக்கள்

பெயர் (அ) குறிப்பு சொல் இல்லாத ஒரு கட்டுரு பெயரற்ற கட்டுரு எனப்படும்

```
struct
{
   long rollno;
   int age;
   float weight;
} student;
```

student என்பது மேலே உள்ள கட்டுருவிற்கு குறிப்பு பெயராக குறிப்பிடப்பட்டுள்ளது. மேலும் அதன் உறுப்புகளை student.rollno, student.age மற்றும் student.weight என அணுக முடியும்.

12.7.4 கட்டுரு உறுப்புகளுக்கு தொடக்க மதிப்பிருத்தல் (Initializing Structure elements)

மாறிகளுக்கு மதிப்புகள் இருத்தப்படுவதை போன்றே கட்டுரு உறுப்புகளுக்கும் மதிப்பு இருத்தப்படும்.

Chapter 12.indd 151 03/09/18 5:08 PM

(எ-கா)

```
balu.rollno= "702016";
balu.age= 18;
balu.weight= 48.5;
```

அணிகளுக்கு மதிப்பு இருத்தப்படுவதை போன்றே இதற்கும் நேரடியாக மதிப்புகளை இருத்தலாம்;

```
balu={702016, 18, 48.5};
```

12.7.5 கட்டுருக்களுக்கு மதிப்பிருத்துதல் (Structure Assignments):

ஒவ்வொரு உறுப்பிற்கும் தனித்தனியாக மதிப்புகளை இருத்துவதற்கு பதிலாக கட்டுருவின் உறுப்புகளுக்கு நேரடியாக மதிப்பிருத்தலாம்.

எடுத்துக்காட்டாக மகேஷ் மற்றும் பிரவிண் இருவரும் ஒரே வயது உயரம் மற்றும் எடை உடையவர்களாக இருந்தால் மகேஷின் மதிப்புகள் முழுவதையும் பிரவீனுக்கு நகலெடுக்க முடியும்.

> இரண்டு கட்டுரு மாறிகள் (அ) பொருள்கள் ஒரே வகையாக இருந்தால் மட்டுமே கட்டுரு மதிப்பிருத்துதலை செய்ய இயலும்.

```
struct Student
{
    int  age;
    float height, weight;
}mahesh;
```

மகேஷின் வயது, உயரம் மற்றும் எடைகள் முறையே 17, 164.5 மற்றும் 52.5 ஆகும்.

கீழ்காணும் கூற்று அதற்கான மதிப்புகளை இருத்தும்.

```
mahesh = {17, 164.5, 52.5};
praveen =mahesh;
```

மேற்கண்ட கூற்றானது praveen என்ற பொருளின் தரவு உறுப்புகளுக்கும் அதே வயது, உயரம் மற்றும் எடை மதிப்புகளை இருத்தும்.

எடுத்துக்காட்டுகள்:

பின்வரும் c++ நிரல் மாணவர்களின் தகவல்களை உள்ளீடாக விசைப்பலகையின் மூலம் பெற்று மற்றும் அதைத் திரையில் காட்டுகிறது.

```
#include <iostream>
using namespace std;
struct Student
{
int  age;
        float height, weight;
} mahesh;
void main( )
{
        cout<< " Enter the age:"<<endl;
        cin>>mahesh.age;
        cout<< "Enter the height:"<<endl;
        cin>>mahesh.height;
        cout<< "Enter the weight:"<<endl;
        cin>>mahesh.weight;
        cout<< "The values entered  for Age, height and weight are"<<endl;
        cout<<mahesh.age<< "\t"<<mahesh.height<< "\t"<<Mahesh.weight;
}
```

வெளியீடு:

```
        Enter the age:
        18
        Enter the height:
        160.5
        Enter the weight:
        46.5
        The values entered  for Age, height and weight are
        18      160.5       46.5
```


Chapter 12.indd 153 03/09/18 5:08 PM

பின்வரும் c++ நிரலானது கட்டுருவின் உறுப்புகளுக்கு மதிப்பினை இருத்தி அவற்றை திரையில் வெளிப்படுத்துகிறது.

```
include<iostream>
using namespace std;
structEmployee
{
char name[50];
int age;
          float salary;
};
int main()
{
          structEmployeee1;
          cout<< "Enter Full name: ";
          cin>>e1.name;
          cout<<endl<<"Enter age: ";
          cin>>e1.age;
cout<<endl<< "Enter salary: ";
          cin>>e1.salary;
          cout<< "\nDisplaying Information." <<endl;
          cout<< "Name: " <<e1.name <<endl;
          cout<<"Age: " <<e1.age <<endl;
          cout<< "Salary: " <<e1.salary;
return 0;
}
```

வெளியீடு:

```
          Enter Full name:
          Ezhil
          Enter age:
           27
          Enter salary:
          40000.00
          Displaying Information.
          Name: Ezhil
          Age: 27
          Salary: 40000.00
```


Chapter 12.indd 154 03/09/18 5:08 PM

12.7.6 பின்னலான கட்டுருக்கள் (Nested Structures)

ஒரு கட்டுரு அமைப்புக்குள் மற்றொரு கட்டுரு அமைந்தால் அது பின்னலான கட்டுருக்கள் எனப்படும். முந்தைய எடுத்துக்காட்டுகளில், மாணவர்களின் தகவல்களை பெற அல்லது இருத்தி வைக்க Student கட்டுருவானது (structure Student) பயன்படுத்தப்பட்டது. Student கட்டுருவில் மாணவர்களின் விவரங்களுடன் பிறந்த தேதி பற்றிய விவரத்தையும் இணைத்து கொள்ள முடியும். மாவணவரின் பிறந்த தேதி விவரத்தில் தேதி, மாதம், மற்றும் வருடம் என மூன்று பகுதிகள் உள்ளன. (எடுத்துக்காட்டு 25-Nov-2017) மாணவர்களின் பிறந்த தேதி பற்றிய விவரத்தை இருத்தி வைக்க மற்றொரு கட்டுருவானது பயன்படுகிறது. பின்வரும் குறிமுறையானது மாணவர்கள் பிறந்த தேதி பற்றிய விவரங்களை கொண்ட கட்டுருவை உருவாக்குகிறது.

```
struct dob

{

int date;

charmonth[3];

int year;

};
```

இந்த கட்டுருவிற்கு மதிப்புகள் பின்வருமாறு இருத்தப்படுகின்றன.

```
dob= {25,"NOV",2017}
```

மாணவரின் பிறந்த தேதியானது Student கட்டுருவில் ஒரு தரவு உறுப்பாக சேர்க்கப்பட வேண்டுமெனில் Student கட்டுருவானது பின்வருமாறு அமைக்கப்படுதல் வேண்டும்

```
struct Student
{
        int  age;
        float height, weight;
        struct dob
{
        int date;
        charmonth[4];
        int year;
};
}mahesh;
void main( )
{
        cout<< " Enter the age:"<<endl;
        cin>>mahesh.age;
        cout<< "Enter the height:"<<endl;
        cin>>mahesh.height;
        cout<< "Enter the weight:"<<endl;
        cin>>mahesh.weight;
        cout<< "The Date of birth:"<<endl;
        cout<< " Enter the day:"<<endl;
```

```
cin>>mahesh.dob.date;
cout<< "Enter the month:"<<endl;
cin>>mahesh.dob.month;
cout<< "Enter the year:"<<endl;
cin>>mahesh.dob.year;
cout<< "The values entered  for Age, height and weightare"<<endl;
cout<<mahesh.age<< "\t"<<mahesh.height<< "\t"<<mahesh.weight<<endl;
cout<< "His date of Birth is:"<<endl<<mahesh.dob.date<<  "-"<<mahesh.dob.
month<< "-"
<<mahesh.dob.year;
```

வெளியீடு:

```
Enter the age:
18
Enter the height:
160.5
Enter the weight:
46.5
The Date of birth
Enter the day:
25
Enter the month:
NOV
Enter the year:
2017
The values entered  for Age, height and weight are
18      160.5           46.5
His date of Birth is:
25-NOV-2017
```

பின்னலான கட்டுருவானது மற்றொரு கட்டுருவுக்கு உறுப்புகளைப் போல செயல்படும்.

12.7.7 கட்டுருக்களின் அணிகள் (Array of Structures)

ஒரு வகுப்பில் பல மாணவர்கள் இருக்கிறார்கள். எனவே ஒரு மாணவரின் கட்டுரு வரையறையை அனைத்து மாணவர்களுக்கும் விரிவுபடுத்தலாம். ஒரு வகுப்பில் 20 மாணவர்கள் இருந்தால் 20 மாணவர்களுக்கும் தனித்தனியாக 20 கட்டுருக்கள் தேவைப்படும். இந்த தேவைக்காக கட்டுருக்களின் அணி பயன்படுகிறது. உள்ளிணைந்த தரவு வகைளில் ஒரு அணி அறிவிக்கப்பட்டது போன்றே கட்டுருக்களிலும் அணிகள் அறிவிக்க முடியும்.

Chapter 12.indd 156 03/09/18 5:08 PM

பின்வரும் நிரலானது 20 மாணவர்களின் விவரங்களை உள்ளீடாக பெற்று அந்த மாணவர்களின் விவரங்களையும் அச்சிட்டு காட்டும்.

```
#include <iostream>
using namespace std;
struct Student
{
        int  age;
        float height, weight;
        char name[30];
};
void main( )
{
struct Student std[20];
int i;
        cout<< " Enter the details for 20 students"<<endl;
        for(i=0;i<20;i++)
{
        cout<< " Enter the details for student"<<i+1<<endl;
        cout<< " Enter the age:"<<endl;
        cin>>std[i].age;
        cout<< "Enter the height:"<<endl;
        cin>>std[i].height;
        cout<< "Enter the weight:"<<endl;
        cin>>std[i].weight;
}
        cout<< "The values entered  for Age, height and weight are"<<endl;
        for(i=0;i<20;i++)
cout<<"Student "<<i+1<< "\t"<<std[i].age<< "\t"<<std[i].height<< "\t"<<std[i].weight;
}
```


Chapter 12.indd 157 03/09/18 5:08 PM

வெளியீடு:

```
Enter the details for 20 students
Enter the details for student1
Enter the age:
18
Enter the height:
160.5
Enter the weight:
46.5
Enter the details for student2
Enter the age:
18
Enter the height:
164.5
Enter the weight:
61.5
......................
....................
The values entered  for Age, height and weight are
Student 1      18      160.5          46.5
Student 2      18      164.5          61.5
............
```

மேற்கண்ட நிரல் 20 மாணவர்களின் வயது, உயரம் மற்றும் எடையை உள்ளீடாக பெற்று அதே விவரங்களை வெளியீடாக அச்சிட்டது. இடம் கட்டுப்பாட்டின் காரணமாக வெளியீடு இரண்டு மாணவர்களுக்கு மட்டுமே காட்டப்பட்டுள்ளது.

ஒரு நிறுவனத்தில் மூன்று பணியாளர்கள் உள்ளனர். நாம் அவர்களின் விவரங்களை உள்ளீடாக பெற்று அனைத்து விவரங்களையும் அச்சிட விரும்பினால் இந்த நிறுவனத்தின் பணியாளருக்கான கட்டுருவை கட்டுருக்களின் அணியாக கொடுக்க வேண்டும். பணியாளர்களின் கட்டுருவை கட்டுருக்களின் அணியாக அறிவிப்பதன் மூலம் இதை செய்ய முடியும்.

```
include<iostream>
using namespace std;
struct Employee
{
```

```
char name[50];
int age;
float salary;
};
int main()
{
struct Employee e1[3];
int i;
cout<< “Enter the details of 3 employees”<<endl;
for(i=0;i<3;i++)
{
        cout<< “Enter the details of Employee ”<< i+1<<endl;
        cout<< "Enter name: ";
        cin>>e1[i].name;
        cout<<endl<<"Enter age: ";
        cin>>e1[i].age;
        cout<<endl<< "Enter salary: ";
        cin>>e[i]1.salary;
}
        cout<< "Displaying Information" <<endl;
        for(i=0;i<3;i++)
{
        cout<< “ The details of  Employee” <<i+1<<endl;
        cout<< "Name: " <<e1[i].name <<endl;
        cout<<"Age: " <<e1[i].age <<endl;
        cout<< "Salary: " <<e1[i].salary;
return 0;
}
```

வெளியீடு:

```
Enter the details of 3 employees
Enter the details of Employee 1
Enter name:
Lilly
Enter age:
42
Enter salary:
40000.00
```


Chapter 12.indd 159 03/09/18 5:08 PM

```
Enter the details of Employee 2
Enter name:
Aster
Enter age:
38
Enter salary:
60000.00
Enter the details of Employee 3
Enter name:
Jasmine
Enter age:
45
Enter salary:
80000.00
Displaying Information.
The details of  Employee 1
Name:Lilly
Age: 42
Salary: 40000.00
The details of  Employee 2
Name:Aster
Age:38
Salary:60000.00
The details of  Employee 3
Name:Jasmine
Age:45
Salary:80000.00
```

12.7.8 செயற்கூறினுக்கு கட்டுருக்களை அனுப்பி வைத்தல் (Passing Structures to functions)

உள்ளிணைந்த தரவின மாறிகளை செயற்கூறுக்கு அளபுருவாக அனுப்பி வைப்பதை போன்றே கட்டுருவின் பொருளையும் அளபுருவாக அனுப்ப முடியும்.

கட்டுருவே அளபுருவாக அனுப்பப்பட்டால் அது மதிப்பு மூலம் அழைத்தல் எனப்படும்.

கட்டுருக்களின் குறிப்பு அளபுருக்களாக அனுப்பப்பட்டால் அது "குறிப்பு மூலம் அழைத்தல்"எனப்படும்.

1. மதிப்பு மூலம் அழைத்தல் (Call by value)

மதிப்பு மூலம் அழைத்தல் முறையில் கட்டுருவானது செயற்கூறுக்கு அளபுருவாக அனுப்பப்படும் போது செயற்கூறினுள் கட்டுருவின் தரவு உறுப்புகளன் செய்யப்படும் எந்த மாற்றமும் அளபுருவாக அனுப்பப்பட்ட கட்டுரு மாறியின் தரவு உறுப்புகளின் பாதிப்பை ஏற்படுத்தாது.

Chapter 12.indd 160 03/09/18 5:08 PM

இந்த உதாரணத்தை கவனியுங்கள்:

```
#include <iostream>
using namespace std;
struct Employee
{
char name[50];
int age;
float salary;
};
void printData(Employee);   // Function declaration
int main()
{
        Employee p;
        cout<< "Enter Full name: ";
        cin>>p.name;
        cout<< "Enter age: ";
        cin>>p.age;
        cout<< "Enter salary: ";
        cin>>p.salary;
        // Function call with structure variable as an argument
printData(p);
return 0;
}
void printData(Employee p)
{
        cout<< "\nDisplaying Information." <<endl;
        cout<< "Name: " << p.name <<endl;
        cout<<"Age: " <<p.age<<endl;
        cout<< "Salary: " <<p.salary;
}
```

வெளியீடு:

```
Enter Full name: Kumar
Enter age: 55
Enter salary: 34233.4
Displaying Information.
Name: Kumar
Age: 55
Salary: 34233.4
```


Chapter 12.indd 161 03/09/18 5:08 PM

மேற்கண்ட உதாரணத்தில் Employee என்ற கட்டுரு அறிவிக்கப்பட்டு பயன்படுத்தப்படுகிறது. கட்டுருவில் உள்ளிட்ட (அ) மதிப்புகளான பணியாளரின் பெயர், வயது மற்றும் சம்பளம் ஆகியவை printData() என்ற ஒரு செயற்கூறினை பயன்படுத்தி காட்டப்படுகின்றன.மேற்கூறப்பட்ட செயற்கூறின் அளபுரு என்பது கட்டுரு Employee கட்டுருவின் பொருள் ஆகும். readData() என்ற செயற்கூறின் மூலம் உள்ளீட்டை பெறமுடியும்.

```
#include <iostream>
using namespace std;
struct Employee {
char name[50];
int age;
float salary;
};
Employee readData(Employee);
voidprintData(Employee);
int main()
{
        Employee p;
        p = readData(p);
        printData(p);
return 0;
}
        Employee readData(Employee p) {
        cout<< "Enter Full name: ";
        cin.get(p.name, 50);
        cout<< "Enter age: ";
        cin>>p.age;
        cout<< "Enter salary: ";
        cin>>p.salary;
return p;
}
voidprintData(Employee p)
{
        cout<< "\nDisplaying Information." <<endl;
        cout<< "Name: " << p.name <<endl;
        cout<<"Age: " <<p.age<<endl;
        cout<< "Salary: " <<p.salary;
```

```
}
char name[50];
int age;
float salary;
};
voidreadData(Employee&);
voidprintData(Employee);
int main()
{
        Employee p;
        readData(p);
        printData(p);
return 0;
}
void readData(Employee &p) {
        cout<< "Enter Full name: ";
cin.get(p.name, 50);
        cout<< "Enter age: ";
        cin>>p.age;
        cout<< "Enter salary: ";
        cin>>p.salary;
}
voidprintData(Employee p)
{
        cout<< "\nDisplaying Information." <<endl;
        cout<< "Name: " << p.name <<endl;
        cout<<"Age: " <<p.age<<endl;
        cout<< "Salary: " <<p.salary;
}
```

> கட்டுருக்களானது பொதுவாக குறிப்பு மூலம் அனுப்புதல் முறையில் அளபுருவாக அனுப்பப்படுகின்றன. ஏனெனில் இது குறைவான நினைவகத்தை எடுத்துக்கொண்டு வேகமாக செயல்பட உதவுகிறது.

வெளியீடு:

```
        Enter Full name: Kumar
        Enter age: 55
        Enter salary: 34233.4
        Displaying Information.
        Name: Kumar
        Age: 55
        Salary: 34233.4
```


Chapter 12.indd 163 03/09/18 5:08 PM

```
voidreadData(Employee&);
voidprintData(Employee);
int main()
{
        Employee p;
        readData(p);
        printData(p);
return 0;
}
void readData(Employee &p) {
        cout<< "Enter Full name: ";
cin.get(p.name, 50);
        cout<< "Enter age: ";
        cin>>p.age;
        cout<< "Enter salary: ";
        cin>>p.salary;
}
voidprintData(Employee p)
{
        cout<< "\nDisplaying Information." <<endl;
        cout<< "Name: " << p.name <<endl;
        cout<<"Age: " <<p.age<<endl;
        cout<< "Salary: " <<p.salary;
}
```

வெளியீடு:

```
        Enter Full name: Kumar
        Enter age: 55
        Enter salary: 34233.4
        Displaying Information.
        Name: Kumar
        Age: 55
        Salary: 34233.4
```

இந்த நிரலின் வெளியீடானது முந்தைய நிரலின் வெளியீட்டை போன்றே இருக்கும்.

2.குறிப்பு மூலம் அழைத்தல்

இந்த முறையில் ஒரு செயற்கூறினுக்கு அளபுருக்களை அனுப்பும் போது, கட்டுரு மாறி (அ) பொருளின் முகவரியானது செயற்கூறினுக்கு முகவரி செயற்குறியை (& address of operator) பயன்படுத்தி அனுப்பப்படுகிறது. எனவே, செயற்கூறினுள் உள்ள கட்டுரு மாறியின் உறுப்புகளில் எந்த வித மாற்றத்தை செய்தாலும், அது அழைப்புக் கூற்றினுள் பிரதிபலிக்கும்.

பின்வரும் எடுத்துக்காட்டானது குறிப்பு மூலம் swap() செயற்கூறினை அழைக்கிறது.

```
#include<iostream.h>
using namespace std;
struct Employee
{
int Id;
char Name[25];
int Age;
long Salary;
};
        Employee Input();
void main()
{
        Employee e;
        Emp = Input();
        Cout<< “The values Entered are”<<endl:
        cout<< "\n\nEmployee Id : " <<e.Id;
        cout<< "\nEmployee Name : " <<eE.Name;
        cout<< "\nEmployee Age : " <<e.Age;
        cout<< "\nEmployee Salary : " <<e.Salary;
}
        Employee Input()
{
        Employee e;
        cout<< "\nEnter Employee Id : ";
        cin>>e.Id;
        cout<< "\nEnter Employee Name : ";
cin>>e.Name;
        cout<< "\nEnter Employee Age : ";
        cin>>e.Age;
        cout<< "\nEnter Employee Salary : ";
        cin>>e.Salary;
return e;
}
```

வெளியீடு:

```
        Enter Employee Id : 10
        Enter Employee Name : Ajay
        Enter Employee Age : 25
        Enter Employee Salary : 15000
        Employee Id : 10
        Employee Name : Ajay
        Employee Age : 25
        Employee Salary : 15000
```


Chapter 12.indd 165 03/09/18 5:08 PM

3.செயற்கூறிலிருந்து கட்டுருக்களை திருப்பி அனுப்புதல் (Returning Structures from function)

ஒரு கட்டுருவை அதன் பொருள் மூலம் ஒரு செயற்கூறினுக்கு அனுப்பலாம். ஆகையால் ஒரு செயற்கூறுக்கு ஒரு கட்டுருவை அனுப்புவதும் அல்லது ஒரு செயற்கூறுக்கு ஒரு கட்டுருவின் பொருளை அனுப்புவதும் ஒன்றுதான். ஏனெனில் கட்டுருக்களின் பொருள் கட்டுருவை குறிக்கிறது.சாதாரண மாறிகளை போன்றே கட்டுரு மாறிகளையும்(கட்டுரு பொருள்கள்) மதிப்பு அல்லது குறிப்புகள் (முகவரிகள்) மூலம் அனுப்ப முடியும்.

நினைவில் கொள்க

- கட்டுரு என்பது பல்வேறு வகையான தரவின உறுப்புகளைக் கொண்ட பயனரால் வரையறுக்கப்பட்ட தரவினம் ஆகும்.
- கட்டுருவை வரையறுக்க stuct என்னும் சிறப்பு சொல் பயன்படுத்தப்படுகிறது.
- கட்டுரு உறுப்புகளை குறிப்பிட பொருளின் பெயர் மற்றும் உறுப்பினரின் பெயருக்கு இடையில் ஒரு புள்ளி (.) இயக்கி (Dot Operator) பயன்படுத்தப்படுகிறது.
- பெயர் அல்லது குறிப்பு சொல் இல்லாத ஒரு கட்டுரு பெயரற்ற கட்டுரு எனப்படும்.
- தனியான மதிப்பிருத்து கூற்றுகளை பயன்படுத்தி கட்டுரு உறுப்புகளை தொடங்கி வைக்கலாம். (அ) மதிப்புகளை அடைப்பு குறிகளுக்குள் தந்து அறிவிக்கும் நேரத்திலும் தொடங்கி வைக்க முடியும்.
- ஒரு கட்டுருவை மற்றொரு கட்டுருவினுள் அறிவிப்பதே பின்னலான கட்டுரு எனப்படும்.
- இரண்டு பொருள்களும் ஒரே கட்டுரு வகையாக இருக்கும் போது மட்டுமே ஒரு கட்டுரு பொருளை மற்றொரு கட்டுருக்குள் இருத்தலாம்.
- ஒரு கட்டுருவானது அணியை அதன் உறுப்பு கூறுகளாக கொண்டிருக்கலாம்.
- கட்டுரு மாறி அல்லது பொருளுக்கும் அணிகளை உருவாக்க முடியும்.
- செயற்கூறினுக்கு கட்டுருக்களை மதிப்பு மூலம் அழைத்தல் அல்லது குறிப்பு மூலம் அழைத்தல் முறையில் அனுப்பப்படுகிறது. கட்டுருக்கள் அல்லது அதன் குறிப்புகளை செயற்கூறுகள் திருப்பி அனுப்பும்.

மதிப்பீடு

பகுதி அ

N1R5E4

I சரியான விடை யை தேர்வு செய்யவும் :

1. கட்டுருக்களின் தரவு உறுப்புகள் எவ்வாறு அழைக்கப்படும்

   (அ) பொருள்கள் (ஆ) உறுப்புகள்

   (இ) தரவு (ஈ) பதிவுகள்

2. கட்டுரு வரையறை எந்த செயற்குறியுடன் முடிவடைதல் வேண்டும்?

   (அ) : (ஆ) }

   (இ) ; (ஈ) ::

3. கட்டுருக்களை அறிவிக்கும் போது என்ன ஏற்படும்?

(அ) அது எந்த நினைவகத்தையும் ஒதுக்காது

(ஆ) அது நினைவகத்தை ஒதுக்கும்

(இ) அது அறிவிக்கும் மற்றும் தொடங்கும்

(ஈ) அது அறிவிக்க மட்டும் செய்யும்

4. இந்த நிரலின் வெளியீடு என்ன?

```
#include <iostream>
#include <string.h>
using namespace std;
int main()
{
struct student
{
int n;
char name[10];
};
student s;
s.n = 123;
strcpy(s.name, "Balu");
cout<<s.n<<endl;
cout<< s.name <<endl;
return 0;   }
```

அ) 123Balu (ஆ)BaluBalu (இ) Balu123 (ஈ) 123Balu

5. ஒரு கட்டுரு அறிவிப்பு கீழ்கண்டவாறு கொடுக்கப்பட்டுள்ளது.

```
struct Time
{
    int hours;
    int minutes;
```

```
int seconds;
}t;
```

மேலே உள்ள அறிவிப்பில் seconds என்ற கட்டுரு மாறியை பின் வருவனவற்றுள் எது குறிக்கிறது?

அ) Time.seconds (ஆ) Time::seconds (இ)seconds (ஈ) t. seconds

6. இந்த நிரலின் வெளியீடு என்னவாக இருக்கும்?

```
#include <iostream>
using namespace std;
structShoeType
    {
stringname;
double price;
    };
int main()
{
ShoeTypeshoe1, shoe2;
shoe1.name = "Adidas";
shoe1.price = 9.99;
cout<< shoe1.name<< " # "<< shoe1.price<<endl;
shoe2 = shoe1;
    shoe2.price = shoe2.price / 9;
cout<< shoe2.name<< " # "<< shoe2.price;
return 0;
  }
```

(a) Adidas # 9.99

Adidas # 1.11 (b) Adidas # 9.99

Adidas # 9.11 (c) Adidas # 9.99

Adidas # 11.11 (c) Adidas # 9.11

Adidas # 11.11

7. கீழ்கண்டவற்றுள் எவை சரியான கட்டுரு வரையறை?

(அ) struct {int num;} (ஆ) struct sum {int num;}

(இ) struct sum int sum; (ஈ)struct sum {int num;};

8. ஒரு கட்டுரு வரையறை கீழே கொடுக்கப்பட்டுள்ளது.

```
struct employee
  {
intempno;
charename[10];
}e[5];
```

மேற்கண்ட அறிவிப்புகளை பயன்படுத்தும் போது இதில் எது சரியான கூற்று?

(அ) cout<<e[0].empno<<e[0].ename; (ஆ) cout<<e[0].empno<<ename;

(இ)cout<<e[0]->empno<<e[0]->ename; (ஈ) cout<<e.empno<<e.ename;

9. கீழ்கண்டவற்றுள் எது கட்டுருவின் உறுப்பு அல்ல?

(அ) Another structure (ஆ) Function

(இ) Array (ஈ) variable of double datatype

10. கட்டுரு உறுப்புகளை அணுகும் போது புள்ளி செயற்குறியின் வலது புறமுள்ள குறிப்பெயரின் பெயர்

(அ) structure variable (ஆ) structure tag

(இ) structure member (ஈ) structure function

பகுதி– ஆ

II குறுவினாக்கள் (2 மதிப்பெண்கள்)

1. வரையறு-கட்டுரு. அதன் பயன் என்ன?

2. 100 முழு எண்களை தேக்கி வைக்க கீழ்கண்டவற்றுள் எதை பயன்படுத்துவது நல்லது?

   அணி (அ) கட்டுரு. காரணம் கூறுக.

3. பின்வரும் கட்டுரு வரையறையில் பிழை என்ன?

```
struct   employee{ inteno;charename[20];char dept;}
Employee e1,e2;
```

4. மாணவர் கட்டுருவில் தேர்வு எண், மாணவர் பெயர் மற்றும் 5 பாடங்களின் மதிப்பெண்களை அணியில் சேமிக்கும் மாணவர் கட்டுருவிற்கான கட்டுரு வரையை எழுதுக.

Chapter 12.indd 169 03/09/18 5:08 PM

5. ஒரு செயற்கூறினுக்கு கட்டுருவை அனுப்பும் போது ஏன் குறிப்பு மூலம் அழைத்தல் சிறந்தது?

6. நிரலானது dev C++ -ல் தொகுக்கப்பட்டால் கீழ்கண்ட உயர்த்திக்காட்டப்பட்டுள்ள மாறியின் பைட் அளவு என்ன?

   struct A{ float f [3]; char ch[5];long double d;};

   struct B{ A a; int arr[2][3];}b[3];

7. பின்வரும் நிரல் தொகுதி முழுவதும் சரியானதா? பிழை இருந்தால் அடையாளம் காண்க?

   struct sum1{ int n1,n2;}s1;

   struct sum2{int n1,n2}s2;

   cin>>s1.n1>>s1.n2;

   s2=s1;

8. அணி மற்றும் கட்டுருவை வேறுபடுத்துக.

9. கட்டுரு உறுப்புகளை தொடங்குவதற்கான பல்வேறு விதமான வழிகள் யாவை?

10. கீழ்க்கண்ட c++ வரையறையில் என்ன மாதிரியான பிழை உள்ளது?

    A. struct point (double x, y)

    B. struct point { double x, double y };

    C. struct point { double x; double y }

    D. struct point { double x; double y; };

    E. struct point { double x; double y; }

பகுதி– இ

III சிறுவினாக்கள் (3 மதிப்பெண்கள்)

1. ஒரு செயற்கூறுக்கு எவ்வாறு கட்டுருவை அனுப்ப முடியும் ?

2. பின்வரும் குறிமுறையானது S என்ற எழுத்தில் தொடங்கும் பெயரைக்கொண்ட அனைத்து மாணவர்களின் மொத்த மதிப்பெண்களின் கூட்டு தொகையை கணக்கிட்டு திரையில் காட்டுகிறது. இதற்கு தேவையான விடுபட்ட கூற்றுகளை நிரப்பவும்.

```
struct student {intexamno,lang,eng,phy,che,mat,csc,total;char name[15];};
int main()
{
student s[20];
for(int i=0;i<20;i++)
```

```
{
        …………………….. //accept student details
        }
        for(int i=0;i<20;i++)
        {
        …………………….. // "S" என்ற எழுத்துடன் தொடங்கும் பெயரை சரிபார்க்கவும்
        ……………………. // சரிபார்த்த பெயரை திரையில் காட்டவும்.
        }
        return 0;
        }
```

3. பின்னலான கட்டுரு என்றால் என்ன? எடுத்துக்காட்டு தருக.

4. தொடரியல் பிழைகளை நீக்கிய பிறகு கீழ்கண்ட நிரலை மாற்றி எழுதவும். ஒவ்வொரு பிழையை கண்டறிந்து அடிக்கோடிடுங்கள்.

   Underline each correction.

```
struct movie
{
charm_name[10];
charm_lang[10];
floatticketcost =50;};
Movie;
void main()
{
gets(m_name);
cin>>m_lang;
return 0;
}
```


Chapter 12.indd 171 03/09/18 5:08 PM

5. பின்வரும் இரண்டு நிரல்களுக்கும் இடையே உள்ள வேறுபாடு என்ன?

(a)
```
#include <iostream.h>
struct point { double x; double y; };
int main() {
struct point test;
test.x = .25; test.y = .75;
cout<<test.x<<test.y;
return 0;
}
```

(b)
```
#include <iostream.h>
struct { double x; double y; } Point;
int main(void) {
Point test={.25,.75};
return 0;
}
```

6. ஒரு கட்டுருவின் உறுப்புகளை எவ்வாறு அணுக முடியும்? எடுத்துக்காட்டு தருக.

7. கட்டுருவை அறிவித்தலுக்கான தொடரியலை எழுது. எடுத்துக்காட்டு கொடு.

8. பின்வரும் கட்டமைப்பு வரையறைக்கு விசைப்பலகையின் மூலம் தரவை பெருவதற்கு பயனர் வரையறைக்கும் செயற்கூறை எழுதவும்.
```
struct date{ intdd,mm,yy};
struct item { int itemid;char name[10];float price;date date_manif;}
```

9. பெயரற்ற கட்டுரு என்றால் என்ன? எடுத்துக்காட்டு தருக.

10. விசைப்பலகையின் மூலம் மதிப்பினை பெற்ற பிறகு கட்டுருவை திருப்பி அனுப்புவதற்கான பயனர் வரையறுக்கும் செயல் கூறை எழுதுக.
```
struct Item{int itemno;float price;};
```

பகுதி – ஈ

IV பெருவினாக்கள் (5 மதிப்பெண்கள்)

1. கட்டுருக்களின் அணியை எடுத்துக்காட்டுடன் விளக்குக.

2. கட்டுருவில் மதிப்பு மூலம் அழைத்தலை விவரி.

Chapter 12.indd 172 03/09/18 5:08 PM

3. செயற்கூறுக்கு கட்டுருக்களை அனுப்பும் போது குறிப்பு மூலம் அழைத்தல் எவ்வாறு பயன்படுகிறது எடுத்துக்காட்டு தருக?

4. பின்வரும் கட்டுரு வரையறையை பயன்படுத்தி இரண்டு தூரங்களை (distance) கூட்டுவதற்கான c++ நிரலை எழுதுக.

```
struct Distance{
int feet;
float inch;
}d1 , d2, sum;
```

5. செயற்கூறினுக்கு கட்டுருவை அனுப்பி இரண்டு கலப்பு எண்களை கூட்டுவதற்கான c++ நிரலை எழுதுக. பின்வரும் கட்டுரு வரையரையை பயன்படுத்துக.

```
struct complex
{
float real;
float imag;
};
```

The prototype of the function is

```
complexaddComplexNumbers(complex, complex);
```

6. book என்ற பெயருடன் கூடிய ஒரு கட்டுருவை அறிவித்து அக்கட்டுருவானது புத்தகத்தின் பெயர், ஆசிரியர் மற்றும் விலை ஆகியவற்றைக் கொண்டிருக்குமாறு அமைக்கவும். அவற்றில் ஆசிரியர் மற்றும் புத்தகத்தின் பெயரானது 20 உறுப்புகளை கொண்ட எழுத்துரு வகையாகவும் ஒவ்வொரு புத்தகத்தின் விலை முழு எண் தரவு வகையாகவும் அறிவிக்கவும். ஒவ்வொரு புத்தகத்தின் பெயர், ஆசிரியர் பெயர், விலை விவரம் ஆகியவற்றை ஏற்கவும். புத்தகத்தின் விவரம் மற்றும் மொத்த விலையை கணக்கிட்டு காண்பிக்க ஒரு பயனர் வரையறுத்த செயற்கூறுவை அழைக்கவும். அழைக்கப்படட்ட செயல்கூறு மொத்த விலையை திருப்பி அனுப்பவும் c++ நிரலை எழுதுக.

7. பேராசிரியர்களை கொண்ட அணிகளை அறிவித்து ஏற்கவும். கணிப்பொறி துறையின் விவரங்கள் மற்றும் A என்ற எழுத்தில் தொடங்கும் பேராசிரியர்களின் பெயர்களை காண்பிக்கவும். கல்லூரி கட்டுருவானது பின்வரும் உறுப்புகளை கொண்டிருக்க வேண்டும். prof_id ஆனது முழு எண் வகையாகவும் பெயர் மற்றும் துறையை எழுத்துருக்களின் வகையாக கொண்ட அணியாக அறிவிக்கவேண்டும்.

8. பின்வரும் c++ நிரலின் வெளியீட்டை எழுதுக.

```
#include<iostream>
#include<stdio>
```

```
#include <string>
#include<conio>
using namespace std;
struct books {
char name[20], author[20];
} a[50];
int main()
{
clrscr();
cout<< "Details of Book No " << 1 << "\n";
cout<< "------------------------\n";
cout<< "Book Name :"<<strcpy(a[0].name,"Programming  ")<<endl;
cout<< "Book Author :"<<strcpy(a[0].author,"Dromy")<<endl;
cout<< "\nDetails of Book No " << 2 << "\n";
cout<< "------------------------\n";
cout<< "Book Name :"<<strcpy(a[1].name,"C++programming" )<<endl;
cout<< "Book Author :"<<strcpy(a[1].author,"BjarneStroustrup ")<<endl;
cout<<"\n\n";
cout<< "================================================\n";
cout<< " S.No\t| Book Name\t|author\n";
cout<< "=======================================================";
for (int i = 0; i < 2; i++) {
  cout<< "\n  " << i + 1 << "\t|" << a[i].name << "\t| " << a[i].author;
   }
cout<< "\n================================================";

return 0;
}
```


Chapter 12.indd 174 03/09/18 5:08 PM

9. பின்வரும் c++ நிரலின் வெளியீட்டை எழுதுக.

```
#include <iostream>
#include <string>
using namespace std;
struct student
  {
          introll_no;
          char name[10];
          longphone_number;
  };

int main(){
student p1 = {1,"Brown",123443};
student p2, p3;
p2.roll_no = 2;
strcpy(p2.name ,"Sam");
p2.phone_number = 1234567822;
p3.roll_no = 3;
strcpy(p3.name,"Addy");
p3.phone_number = 1234567844;
cout<< "First Student" <<endl;
cout<< "roll no : " << p1.roll_no <<endl;
cout<< "name : " << p1.name <<endl;
cout<< "phone no : " << p1.phone_number <<endl;
cout<< "Second Student" <<endl;
cout<< "roll no : " << p2.roll_no <<endl;
cout<< "name : " << p2.name <<endl;
cout<< "phone no : " << p2.phone_number <<endl;
```


Chapter 12.indd 175 03/09/18 5:08 PM

```
cout<< "Third Student" <<endl;
cout<< "roll no : " << p3.roll_no <<endl;
cout<< "name : " << p3.name <<endl;
cout<< "phone no : " << p3.phone_number <<endl;
return 0;
}
```

10. பின்வரும் நிரலில் உள்ள பிழைகளை திருத்துக.

```
#include <istream.h>
structPersonRec
{
charlastName[10];
chaefirstName[10];
int age;
}
PersonRecPeopleArrayType[10];
voidLoadArray(PeopleRecpeop);
void main()
{
PersonRecord people;
for (i = 0; i < 10; i++)
{
cout<<people.firstName<< ' ' <<people.lastName
<<setw(10) <<people.age;
}
}
LoadArray(PersonRecpeop)
{
```


Chapter 12.indd 176 03/09/18 5:08 PM

```
for (int i = 0; i < 10; i++)
{
cout<< "Enter first name: ";
cin<<peop[i].firstName;
cout<< "Enter last name: ";
cin>>peop[i].lastName;
cout<< "Enter age: ";
cin>> people[i].age;}
```

துணை நூல்கள்:

(1) Object Oriented Programming with C++ (4th Edition), Dr. E. Balagurusamy, Mc.Graw Hills.

(2) The Complete Reference C++ (Forth Edition), Herbert Schildt.Mc.Graw Hills.

(3) Computer Science with C++ (A text book of CBSE XI and XII), SumitaArora, DhanpatRai& Co.

(4) The C++ Programming Language,BjarneStroustrup

(5) https://www.tutorialspoint.com

(6) http://www.cs.princeton.edu

(7) https://www.programiz.com

Chapter 12.indd 177 03/09/18 5:08 PM

N1U531

அறிமுகம் – பொருள்நோக்கு நிரலாக்க நுட்பங்கள்

கற்றலின் நோக்கங்கள்

இந்த பாடத்தைப் படித்தபின்னர் மாணவர்கள்.

- பொருள் நோக்கு நிரலாக்க கருத்துருவை பற்றி அறிந்து கொள்ளுதல்.
- செயல்முறை, கூறுநிலை மற்றும் பொருள் நோக்கு நிரலாக்கம் இடையேயான வேறுப்பாட்டை தெரிந்துக் கொள்ளுதல்.
- பொருள் நோக்கு நிரலாக்கத்தின் நன்மைகள், தீமைகள் பற்றி அறிதல்.

13.1 அறிமுகம்

இனக்குழுக்கள் மற்றும் பொருள்கள் அடிப்படையாகக் கொண்ட நிரல் அணுகு முறையை விவரிக்க பொருள் நோக்கு நிரலாக்க கருத்துரு உதவுகின்றன. பொருள் நோக்கு கருத்தியல், தரவு மற்றும் பண்பியல்புகளைக் கொண்ட பொருள்களின் தொகுப்பை மென்பொருளாக அமைக்க அனுமதிக்கிறது. இது தளர்வாக இணைக்கப்பட்ட தரவு மற்றும் பண்பியல்புகளைக் கொண்ட மரபுசார் செயற்கூறு நிரலாக்கப் பயிற்சிக்கு மாறுபட்டது.

1980 – களில் பொருள் என்ற சொல் நிரலாக்க மொழிகளில் தொடர்புபடுத்தப்பட்டது. 1990-களில் உருவாக்கப்பட்ட பெரும்பாலான மொழிகள் பொருள்நோக்கு சிறப்பியல்புகளை பெற்றுள்ளன. இந்த பாடமானது பொருள் நோக்கு நிரலாக்க கருத்தியல்களை அறிமுகப்படுத்துகிறது.

13.2 நிரலாக்க கருத்தியல்கள்

கருத்தியல் என்பது நிரலின் கோட்பாடுகளை ஒழுங்குபடுத்தி அமைத்தலாகும். இது ஒரு நிரலாக்க அணுகுமுறையாகும். கணினியைப் பயன்படுத்தி சிக்கல்களைத் தீர்க்க பல்வேறு அணுகுமுறை உள்ளன. அவை நடைமுறை நிரலாக்கம், கட்டக நிரலாக்கம் மற்றும் பொருள்நோக்கு நிரலாக்கம் ஆகும்.

13.2.1. நடைமுறை நிரலாக்கம் (Procedural programming)

நடைமுறை நிரலாக்கம் என்பது கணிப்பொறிக்கு கட்டளைகளின் பட்டியல்களைக் கொடுத்து, ஒவ்வொரு கட்டளைகளையும் ஏதேனும் ஒரு செயலை செய்யுமாறு கூறுவதாகும். இது விதிமுறைகளின் மீது அதிக கவனம் செலுத்துகிறது. வேலை செய்வதற்கு அழுத்தம் கொடுக்கிறது.

- நடைமுறை நிரலின் முக்கிய சிறப்பம்சங்கள் **(Important features)**
- நிரலானது, துணை நிரல் கூறுகளாகவோ கூறு அல்லது துணை நிரல்களாகவோ கட்டமைக்கப்படுகிறது.
- அனைத்துத் தரவு உறுப்புகளும் முழு தளாவியவை ஆகும்.
- சிறிய அளவிலான மென்பொருள் பயன்பாட்டிற்கு பொருத்தமானது.
- நிரல் குறிமுறைகளைப் பராமரித்தலும் மேம்படுத்தலும் கடினமாகும். ஒரு மாறியின் தரவு வகைகளை மாற்ற வேண்டுமெனில், அதே தரவு வகையைப் பயன்படுத்தும் துணை நிரல்கள் அனைத்திற்கும் அந்த தரவு வகை மாற்றத்தைசெய்யவேண்டியது அவசியமாகும்.

Chapter 13.indd 178 03/09/18 5:16 PM

இது அதிக நேரத்தை எடுத்துக்கொள்கிறது. எடுத்துக்காட்டு:- FORTRAN மற்றும் COBOL

13.2.2. கட்டக நிரலாக்கம் (Modular programming)

கட்டக நிரலாக்கம் என்பது கணிப்பொறிக்கு கட்டளைகளின் பட்டியல்களைக் கொடுத்து, ஒவ்வொரு கட்டளைகளையும் ஏதேனும் ஒரு செயலை செய்யுமாறு கூறுகிறது. ஆனால் இந்த கருத்தியலானது பல கூறுகளைக் கொண்டது. ஒவ்வொரு கூறும் தொடர்புடைய செயற்கூறுகளின் அமைப்பாகும். தரவானது செயற்கூறினுள் மறைக்கப்படுகிறது கூறுகளை மாற்றியமைப்பதன் மூலம் தரவின் சீரமைப்பை மாற்ற முடியும்.

கூறுநிலை நிரலின் சிறப்பம்சங்கள் (Important features of Modular programming)

- தரவைக் காட்டிலும் நெறிமுறைக்கு முக்கியத்துவம் கொடுக்கிறது.
- நிரலானது தனித்தனி கூறுகளாகப் பிரிக்கப்பட்டுள்ளது.
- ஒவ்வொரு கூறும் ஒன்றுக்கொன்று சார்பற்றது மற்றும் தனித்த உள்ளமை தரவைக் கொண்டிருக்கும்.
- கூறுகள் தனது சொந்த தரவுகளின் மீது மட்டுமல்லாமல் அனுப்பப்படும் பிறத் தரவுகளையும் கொண்டு செயல்படுகிறது.

எடுத்துக்காட்டு : - Pascal மற்றும் C

13.2.3. பொருள் நோக்கு நிரல் (Object oriented programming):-

பொருள் நோக்கு நிரலாக்கக் கருத்தியல் நெறிமுறைகளைக் காட்டிலும் தரவுக்கு முக்கியத்துவம் கொடுக்கிறது. இது இனக்குழுக்கள் மற்றும் பொருள்கள் மூலம் நிரலைச் செயல்படுத்துகிறது.

இனக்குழு (CLASS):

C++ -ன் இனக்குழு ஆனது தரவுகளையும் அதற்கு தொடர்பான செயல்கூறுகளையும் உறைபொதியாக்கம் என்னும் கருத்துருவை பயன்படுத்தி ஒரு தொகுதிக்குள் இணைக்கப்பட்டு உருவாக்கப்படுகிறது. இனக்குழு பயனர் வரையறுக்கும் தரவினமாகும். இனக்குழுவானது ஒரே மாதிரியான பொருள்களின் குழுவைக் குறிக்கிறது.

பொதுவான பண்புகளையும் மற்றும் உறவுநிலைகளையும் பகிரக் கூடிய பொருள்களைக் கொண்ட குழுவை வார்ப்புரு அல்லது செயல்திட்டம் என்று குறிக்கலாம்.

பொருள்கள் (OBJECTS)

பொருள் என்பது தொடர்புடைய செயற்கூறுகள், அச்செயற்கூறுகளுக்கான தரவுகள் ஆகியவற்றைக் கொண்ட ஒரு குழுவாகும். பொருளானது பொருள் நோக்கு நிரலாக்கத்தின் அடிப்படை அலகாகும். பொதுவாக பொருளானது இனக்குழுவிலிருந்து உருவாக்கப்படுகிறது. இனக்குழுவின் சான்றுரு ஆனது இவை இனக்குழு மாறிகள் என்றும் அழைக்கப்படுகிறது.

சில பண்பியல்புகள் மற்றும் தனி சிறப்பான செயல்பாடுகளையும் கொண்ட அடையாளம் காணத்தகு உருப்படி பொருள் என்றழைக்கப்படுகிறது.

பொருள்நோக்கு நிரலாக்கத்தின் சிறப்பம்சங்கள் (Important features of Object oriented programming)

- நெறிமுறைக் காட்டிலும் தரவுக்கே முக்கியத்துவம் கொடுக்கிறது.
- தரவு அருவமாக்கமானது நடைமுறை அருவமாக்கத்துடன் கூடுதலாக அறிமுகப்படுத்தப்பட்டது.
- தரவு மற்றும் அவை தொடர்புடைய செயற்கூறுகள் ஒரு தொகுதிக்குள் குழுவாக இருக்கும்.
- செயல்படுத்தக்கூடிய தரவுகளைக் கொண்டு நிரல்கள் வடிவமைக்கப்படுகிறது.
- ஒரே மாதிரியான அல்லது வேறுபட்ட தரவு

வகைகளுக்கு உறவுநிலையை ஏற்படுத்தலாம்.

எடுத்துக்காட்டு - C++, Java,python

13.3. பொருள் நோக்கு நிரலாக்கத்தின் அடிப்படை கருத்துக்கள்

நடைமுறை மற்றும் கட்டக நிரலாக்கத்தில் உள்ள பின்னடைவுகளை மேம்படுத்த பொருள் நோக்கு நிரலாக்கம் உருவாக்கப்பட்டது. பொருள் நோக்கு நிரலாக்கம் ஆற்றல்மிகு வழியில் மென்பொருள்களை உருவாக்குதலில் பொருள் நோக்கு நிரலாக்கமானது மிகவும் முக்கியமானது என்று ஏற்றுக்கொள்ளப்பட்டுள்ளது.

பொருள் நோக்கு நிரலாக்க வழிமறை பின்வருவனவற்றை ஊக்குவிக்கிறது:

- கூறுநிலையாக்கம் (Modularisation): நிரலானது கூறுகளாக பிரிக்கப்படுகிறது.
- மென்பொருள் மறுபயனாக்கம் (Software re-use): நிரலானது ஏற்கனவே உள்ள அல்லது புதிய கூறுகளைக் கொண்டு தொகுக்கப்படுகிறது.

பொருள் நோக்கு நிரலாக்கத்தின் சிறப்பியல்புகள்

- உறைபொதியாக்கம் (Encapsulation)
- தரவு அருவமாக்கம் (Data Abstraction)
- கூறுநிலையாக்கம் (Modularity)
- பல்லுருவாக்கம் (Polymorphism)
- மரபுரிமம் (Inheritance)

13.3.1 உறைபொதியாக்கம்

தரவுகளையும் செயற்கூறுகளையும் ஒரு பொருள் என்னும் வரையறைக்குள் ஒன்றாகப் பிணைத்துவைக்கும் செயல்நுட்பம் உறைபொதியாக்கம் எனப்படுகிறது. இது அருவமாக்கத்தை செயல்படுத்துகிறது.

13.3.2 தரவு அருவமாக்கம் (Data Abstraction)

அருவமாக்கம் என்பது பின்புல விவரங்களை தெரிவிக்காமல் அவசியமான அம்சங்களை மட்டுமே வெளிப்படுத்துவதைக் குறிக்கும். இனக்குழுவானது அருவமாக்க கருத்துருவை வரையறுக்கப்பட்ட பண்புக்கூறுகள் மற்றும் அப்பண்புக்கூறுகளின் மீது செயல்படும் செயற்கூறுகளைக் கொண்டு வரையறுக்கிறது. இது அவசியமான பண்புகளை உருவாக்கப்படும் ஒரு பொருளுக்குள் மறைத்து வைக்கிறது. இந்த பண்புக்கூறுகள் தரவு உறுப்புகளாகும். ஏனெனில் அவை தரவை இருத்தி வைக்கிறது. இந்த தரவுகளின் மீது செயல்படும் செயற்கூறுகள் வழிமுறைகள் அல்லது உறுப்பு செயற்கூறுகள் எனப்படும்.

13.3.3 கூறுநிலைநிலையாக்கம் (Modularity)

கூறுநிலை என்பது ஒரு அமைப்பை பல செயல்பாட்டுத் தொகுதிகளாக (கூறுகள்) பிரித்து பின்னர் அவற்றைத் தொகுத்து பெரிய பயன்பாடாக வடிவமைக்கிறது.

13.3.4 பல்லுருவாக்கம்(Polymorphism)

வேறுபட்ட செய்திகளுக்கு மாறுபட்டுச் செயல்படும்ஒரு பொருளின் திறனே பல்லுருவாக்கம் என்றழைக்கப்படுகிறது.

13.3.5 மரபுரிமம்:

மரபுரிமம் என்பது ஏற்கனவே இருக்கும் இனக்குழுகளின் அடிப்படையில் புதிய இனக்குழுவை (தருவிக்கப்பட்ட இனக்குழு) உருவாக்கும் செயல்முறையாகும். இதன் முக்கிய பயனானது நிரல் குறிமுறை மறுபயனாக்கமாகும்.

13.4 பொருநோக்கு நிரலாக்கத்தின் பலன்கள்(Advantages of OOP)

மறுபயனாக்கம் (Re-usability):

“ஒரு முறை எழுதுதல் பலமுறை பயன்படுத்துதல்” இனக்குழு பயன்படுத்தி இதை நிறைவேற்றலாம்.

மிகைமை (Redundancy):

மரபுரிமம் தரவு மிகைமைக்கும் சிறந்த சான்றாகும். பல இனக்குழுக்களுக்கு தேவையான ஒரே செயல்பாட்டை ஒரு பொது இனக்குழுவின் மூலம் வரையறுத்து அவற்றை

மரபுரிமம் தருவிக்கப்பட்ட இனக்குழுவில் மூலம் தருவித்துக்கொள்ளலாம்.

எளிய பராமரிப்பு (Easy Maintenance):

ஏற்கனவே இருக்கும் குறிமுறையில் சிறிய மாற்றங்களைச் செய்து புதிய பொருளை உருவாக்க முடியும் மேலும் இதை பராமரிப்பதும் மாற்றங்கள் செய்வதும் எளிது.

பாதுகாப்பு (Security):

தரவு மறைப்பு மற்றும் அருவமாக்கம் தேவையான தரவுகளை மட்டும் கொடுப்பதால் தரவு பாதுகாப்பு பராமரிக்கப்படுகிறது.

13.5 பொருள் நோக்கு நிரலாக்கத்தின் தீமைகள் (Disadvantages of OOP)

அளவு (Size)

பொருள் நோக்கு நிரலானது மற்ற நிரல்களை விட அளவில் பெரியது.

உழைப்பு (Effort)

பொருள் நோக்கு நிரலை உருவாக்குவதற்கு அதிக உழைப்பு தேவைப்படுகிறது.

வேகம் (Speed)

பொருள் நோக்கு நிரல்கள் அதிக அளவின் காரணமாக பிற நிரல்களை விட மெதுவாக செயல்படுகிறது.

நினைவில் கொள்க

- கருத்தியல் என்பது நிரலின் கோட்பாடுகளை அமைத்தலாகும். இது ஒரு நிரலாக்க அணுகுமுறையாகும்.
- நடைமுறை நிரல் அல்லது கூறுநிலை நிரல் என்பது கட்டளைகளின் பட்டியல்களைக் கொடுத்து, ஒவ்வொரு கட்டளைகளையும் ஏதேனும் ஒரு செயலை செய்யுமாறு உதவுகிறது.
- நடைமுறை நிரல் விதிமுறைகளின் மீது அதிக கவனம் செலுத்துகிறது. இதில் நிரல்களை, துணை நிரல் கூறு அல்லது துணை நிரல் வடிவத்தில் அமைக்கிறது.
- கூறுநிலை நிரல் ஆனது தொடர்புடைய விதிமுறைகளை கூறுகளில் சேர்த்து கூறுகளில் தரவை மறைக்கிறது.
- பொருள் நோக்கு நிரலாக்கத்தில் கருத்தியல் நெறிமுறைகளைக் காட்டிலும் தரவுக்கு முக்கியத்துவம் கொடுக்கிறது. இது இனக்குழுக்கள் மற்றும் பொருள்கள் மூலம் நிரலைச் செயல்படுத்துகிறது.
- இனக்குழு பயனர் வரையறுக்கும் தரவினமாகும். இனக்குழுவானது ஒரே மாதிரியான பொருள்களின் குழுவைக் குறிக்கிறது.
- பொருளானது பொருள் நோக்கு நிரலாக்கத்தின் அடிப்படை அலகாகும். பொருள் என்பது தொடர்புடைய செயற்கூறுகள், அச்செயற்கூறுகளுக்கான தரவுகள் ஆகியவற்றைக் கொண்ட ஒரு குழுவாகும்.
- தரவுகளையும் செயற்கூறுகளையும் ஒரு பொருள் வரையறை ஒன்றாகப் பினைத்துவைக்கும் செயல்நுட்பம் உறைபொறியாக்கம் எனப்படுகிறது. இது அருவமாக்கத்தை செயல்படுத்துகிறது.

Chapter 13.indd 181 03/09/18 5:16 PM

நினைவில் கொள்க

- அருவமாக்கம் என்பது பின்புல விவரங்களை தெரிவிக்காமல் அவசியமான அம்சங்களை மட்டுமே காட்டுவதைக் குறிக்கும்.
- வேறுபட்ட செய்திகளுக்கு மாறுபட்டுச் செயல்படும் ஒரு பொருளின் திறனே பல்லுருவாக்கம் என்றழைக்கப்படுகிறது.
- மரபுரிமம் என்பது ஏற்கனவே இருக்கும் இனக்குழுகளின் அடிப்படையில் இனக்குழு அடிப்படையில் புதிய இனக்குழுக்களை (தருவிக்கப்பட்ட இனக்குழு) உருவாக்கும் செயல்முறையாகும். இதன் முக்கிய பலன் நிரல் குறிமுறை மறுபயனாக்கமாகும். மரபுரிமம் ஒரு கடத்தியாகும்.

மதிப்பீடு

N1YEQX

பகுதி அ

I சரியான விடை யை தேர்வு செய்யவும் :

1. பின்வருவனவற்றுள் எந்த செயற்கூறு இனக்குழுக்களும் மற்றும் பொருள்களும் அடிப்படையாகக் கொண்ட நிரல் அணுகுமுறையை விவரிக்கிறது ?

   (அ) OOP　　(ஆ) POP

   (இ) ADT　　(ஈ) SOP

2. பின்வருவனவற்றுள் எது இந்த கருத்தியல் விதிமுறைகளின் மீது அதிக கவனம் செலுத்துகிறது ?

   (அ) பொருள்நோக்கி நிரலாக்கம்　　(ஆ) நடைமுறை நிரலாக்கம்

   (இ) கூறுநிலை நிரலாக்கம்　　(ஈ) அமைப்பு நிரலாக்கம்

3. பின்வருவனவற்றுள் எது பயனர் வரையறுக்கும் தரவு வகை?

   (அ) இனக்குழு　　(ஆ)மிதவை

   (இ)முழு எண்　　(ஈ) பொருள்

4. கீழ்கண்டவற்றுள் எது பண்பியல்புகளையும் தனிச் சிறப்பு பண்புகளையும் கொண்ட அடையாளம் காணத்தகு உருப்படி?

   (அ)இனக்குழு　　(ஆ)பொருள்

   (இ)கட்டமைப்பு　　(ஈ)உறுப்பு

5. தரவுகளையும் செயற்கூறுகளையும் ஒரு பொருள் என்னும் வரையறைக்குள் ஒன்றாகப் பினைத்து வைக்கும் செயல்நுட்பம்

   (அ)மரபுரிமம்　　(ஆ) உறைபொதியாக்கம்

   (இ)பல்லுருவாக்கம்　　(ஈ)அருவமாக்கம்

Chapter 13.indd 182　03/09/18 5:16 PM

6. தரவை நிரலின் நேரடி அணுகு முறையிலிருந்து பாதுகாப்பது

(அ) தரவு மறைப்பு (ஆ) உறைபொதியாக்கம்

(இ) பல்லுருவாக்கம் (ஈ) அருவமாக்கம்

7. பின்வருவனவற்றுள் எந்த கருத்துரு ஒரு பொருளின் அவசியமான பண்புகளை உருவாக்கப்படும் பொருளுக்குள் மறைத்து வைக்கிறது?

(அ)இனக்குழு (ஆ) உறைபொதியாக்கம்

(இ)பல்லுருவாக்கம் (ஈ) அருவமாக்கம்

8. பின்வருவனவற்றுள் எது மரபுரிமத்தின் முக்கியமான பண்பாகும்?

(அ) தரவு மறைப்பு (ஆ) உறை பொதியாக்கம்

(இ) குறிமுறை மாற்றம் (ஈ) அணுகுமுறை

9. "ஒருமுறை எழுதுதல் பலமுறை பயன்படுத்துதல் "- அதன் மூலம் நிறைவேற்றப்படுகிறது?

(அ)தரவு மிகைமை (ஆ) மறுபயனாக்கம்

(இ) மாற்றம் (ஈ)தொகுத்தல்

10. எது வெளிப்படைத்தன்மை கொண்ட தரவுகளை உடையது?

(அ) மரபுரிமம் (ஆ) உறைபொதியாக்கம்

(இ) பல்லுருவாக்கம் (ஈ)அருவமாக்கம்

பகுதி– ஆ

II குறுவினாக்கள் (2 மதிப்பெண்கள்)

1. கட்டக நிரலாக்கம் நடைமுறை நிரலாக்க கருத்தியலில் இருந்து எவ்வாறு வேறுபடுகிறது?
2. இனக்குழு மற்றும் பொருள் வேறுபடுத்துக.
3. பல்லுருவாக்கம் என்றால் என்ன?
4. உறைபொதியாக்கம மற்றும் அருவமாக்குதல் எவ்வாறு தொடர்பு படுத்தப்படுகிறது?
5. பொருள் நோக்கு நிரலாக்கத்தின் குறைபாடுகள் யாவை?

பகுதி– இ

III சிறுவினாக்கள் (3 மதிப்பெண்கள்)

1. கருத்தியல் என்றால் என்ன? பல்வேறு வகையான கருத்தியல்களைக் குறிப்பிடுக.
2. நடைமுறை நிரலாக்கத்தின் அம்சங்கள் பற்றி குறிப்பு வரைக.
3. கட்டக நிரலாக்கத்தின் சில அம்சங்களைப் பற்றி பட்டியலிடுக.

4. கூறுநிலையாக்குதல் மற்றும் மென்பொருள் மறு பயனாக்கம் வரையறு.

5. தகவல் மறைப்பு - வரையறு.

பகுதி – ஈ

IV பெருவினாக்கள் (5 மதிப்பெண்கள்)

1. பொருள் நோக்கு நிரலாக்கம் மற்றும் நடைமுறை நிரலாக்கம்- வேறுபடுத்துக.

2. பொருள் நோக்கு நிரலாக்கத்தின் நன்மைகள் யாவை?

3. பொருள் நோக்கு நிரலாக்கத்தை ஆதரிக்கும் அடிப்படைக் கருத்துகளைப் பற்றி குறிப்பு வரைக.

சொற்களஞ்சியம்

கருத்தியல் *(Paradigm)*	நிரலின் கோட்பாடுகளை ஒருங்கினைத்தல்
அருவமாக்கம் *(Abstraction)*	அருவமாக்கம் என்பது பின்புல விவரங்களைத் தெரிவிக்காமல் அவசியமான அம்சங்களை மட்டுமே காட்டுவதைக் குறிக்கும்.
கூறுநிலையாக்கம் *(Modularity)*	கூறுநிலை என்பது ஒரு அமைப்பை பல செயல்பாட்டுத் தொகுதிகளாக (கூறுகள்) பிரித்து பின்னர் அவற்றைத் தொகுத்து பெரிய பயன்பாடாக வடிவமைக்கிறது.
தருவிக்கப்பட்ட இனக்குழு *Base class)*	ஒரு இனக்குழு பண்புகளை புதிதாக உருவாக்கப்படும் இனக்குழு மரபுரிமமாக பெறுவது இது பெற்றோர் இனக்குழு என்றும் அழைக்கப்படுகிறது.
(Derived class)	அடிப்படை இனக்குழுவின் பண்புகளை ஈட்டிக் கொள்ளும் இனக்குழு தருவிக்கப்பட்ட இனக்குழு இது குழந்தை அல்லது துணை இனக்குழு என்றும் அழைக்கப்படுகிறது.
இனக்குழு *(Class)*	இனக்குழு என்பது பொது பண்புகளை பகிரக்கூடிய ஒரேமாதிரியான பொருள்களின் குழுவாகும்.
பொருள் *(Object)*	சில பண்பியல்புகள் மற்றும் சிறப்பு பண்புகளையும் கொண்ட அடையாளம் காணத்தகு உருப்படி.
உரைபொதியாக்கம் *(Encapsulation)*	தரவுகளையும் செயற்கூறுகளையும் ஒரு பொருள் வரையறைக்குள் ஒன்றாகப் பிணைத்துவைக்கும் செயல்நுட்பம்.
மரபுரிமம் *(Inheritance)*	மரபுரிமம் என்பது ஏற்கனவே இருக்கும் இனக்குழுக்களின் அடிப்படையில் (அடிப்படை இனக்குழு) புதிய இனக்குழுக்களை (தருவிக்கப்பட்ட இனக்குழு) உருவாக்கும் செயல்முறை.

Chapter 13.indd 184 03/09/18 5:16 PM

துணை நூல்கள்:

(1) Object Oriented Programming with C++ (4th Edition), Dr. E. Balagurusamy, Mc.Graw Hills.

(2) The Complete Reference C++ (Forth Edition), Herbert Schildt.Mc.Graw Hills.

(3) Computer Science with C++ (A text book of CBSE XI and XII), SumitaArora, DhanpatRai& Co.

(4) A text book of CBSE XI and XII computer science by PreetiArora and Pinky Gupta.

(5) Computer Science with C++ Reeta shoo and Gagansahoo

(6) The C++ Programming Language,BjarneStroustrup

(7) C++ Primer (5th Edition) by S. B. Lippman, J. Lajoie

(8) https://www.tutorialspoint.com

(9) http://www.cs.princeton.edu

(10) https://www.programiz.com

N211FR

இனக்குழுக்கள் மற்றும் பொருள்கள்

கற்றலின் நோக்கங்கள் :

இந்தப்பாடத்தைக் கற்றபின், மாணவர்கள்

- இனக்குழு, பொருள்கள், ஆக்கிகள் மற்றும் அழிப்பியின் பயன்பாடுகளை புரிந்து கொள்ளுதல்.
- ஆக்கி மற்றும் அழிப்பியுடன் கூடிய இனக்குழுவை பயன்படுத்தி C++ நிரல் எழுதுதல்.
- ஆக்கி மற்றும் அழிப்பியுடன் கூடிய C++ நிரலை இயக்கி தவறுகளை திருத்துதல்.

14.1 இனக்குழு அறிமுகம் :

C++ ன் மிக முக்கியமான பண்புக்கூறு இனக்குழு என்பதாகும். இது ஜேர்ன் ஸ்ட்ரௌஸ்ட்ரப் (Bjarne Stroustrup) என்பவரால் உருவாக்கப்பட்டது, முதலில் அவர் சூட்டிய பெயர் "C WITH CLASSES" என்பதாகும். C++ ஆனது, பொருள்நோக்கு நிரலாக்க மொழியின் பொதுவான நான்கு அடிப்படை அம்சங்களை அதாவது அருவமாக்கம், உரைபொதியாக்கம், மரபுரிமம் மற்றும் பல்லுருவாக்கம் ஆகியவற்றை அளிக்கிறது.

14.1.1 இனக்குழுவின் தேவை:

இனக்குழுவானது தரவுகளையும் அவை தொடர்பான செயல் கூறுகளையும் ஒன்றாக சேர்த்துவைக்க வழிசெய்கிறது. இனக்குழுவானது தரவுகளையும், அவற்றிற்கான பண்புகளையும் மட்டும் கொண்டிருக்காமல், நடப்புலகில் நிலவும் பொருளைக் குறிப்பிட உதவுகிறது. மேலும் அப்பொருள் தொடர்பான செயல்பாடுகளையும் கொண்டிருக்கும். இது பயனர் வரையறுக்கும் தரவினத்தை உருவாக்கப் பயன்படுகிறது.

14.1.2 இனக்குழு வரையறை (Declaration of a class)

C++ல் இனக்குழுவின் வரையறையில் Class என்னும் சிறப்புசொல்லைத் (Keywords) தொடர்ந்து இனக்குழுவின் பெயர் இடம்பெற வேண்டும். இனக்குழுவின் உடற்பகுதி வரையறுப்பானது நௌிவு அல்லது கொக்கி வடிவ அடைப்பு ({ }) குறிக்குள் அடைக்கப்பட்டு அரைப்புள்ளியுடன் அல்லது இனக்குழு பொருள்கள் அறிவிப்புடன் முடிவடைதல் வேண்டும்.

குறிப்பு

கட்டுரு மற்றும் இனக்குழுவிற்கு இடையேயான வேறுபாடானது, கட்டுரு உறுப்புகளானது கொடாநிலையாக Public அணுகியல்புடனும் இனக்குழுவின் உறுப்புகளானது Private அணுகியல்புடனும் இருக்கும்.

இனக்குழுவை வரையறுப்பதற்கான பொது வடிவம் :

```
class class.name
        {
        private :
        தரவு உறுப்புகள் அறிவிப்பு ;
        செயல் கூறுகள் அறிவிப்பு ;
        protected :
        தரவு உறுப்புகள் அறிவிப்பு ;
        செயல் கூறுகள் அறிவிப்பு ;
        public :
        தரவு உறுப்புகள் அறிவிப்பு ;
        செயற்கூறுகள் அறிவிப்பு ;
        };
```

- இனக்குழுவின் உடற்பகுதியானது தரவு மாறிகள் மற்றும் செயல்கூறுகளின்

Chapter 14.indd 186 03/09/18 5:49 PM

அறிவிப்பை கொண்டிருக்கும். (தரவு உறுப்புகள் மற்றும் உறுப்பு செயற்கூறுகள்)

- இனக்குழுவின் உடற்பகுதியானது மூன்று அணுகியல்பு வரையறுப்பிகளை கொண்டுள்ளது. (Private, Protected மற்றும் Public)

14.1.3 இனக்குழு அணுகியல்பு வரையறுப்பிகள் :

பொருள்நோக்கு நிரலாக்க மொழியின் மிக முக்கியமான பண்புக்கூறுகளுள் ஒன்று தரவு மறைப்பு (Data hiding) என்பதாகும். இது நிரலில் உள்ள செயற்கூறானது, இனக்குழுவிற்குள் அறிவிக்கப்பட்டிருக்கும் உறுப்புகளை (தரவு உறுப்புகள், உறுப்புச் செயல் கூறுகள்) அணுகுவதற்குத் தடை விதிக்கிறது. இனக்குழுவின் உறுப்புகளை அணுக தடைவிதிப்பதற்கு இனக்குழுவினுள் அறிவிக்கப்பட்டிருக்கும் Private, Protected மற்றும் Public என்ற சிறப்புச் சொற்கள் பயன்படுகிறது.

Private, Protected மற்றும் Public என்ற சிறப்புச் சொற்களானது அணுகியல்பு வரையறுப்பிகள் என அழைக்கப்படுகின்றன. இனக்குழு உறுப்புகளின் கொடாநிலை அணுகியல்பானது Private ஆகும்.

Public உறுப்புகள் :

Public அணுகியல்புடன் கூடிய உறுப்புகளை இனக்குழுவிற்கு வெளியில் இருந்தும் அணுக முடியும். Public என அறிவிக்கப்பட்ட தரவு உறுப்புகளை எவ்வித உறுப்பு செயல்பாடுகளின் உதவியுமின்றி அணுக முடியும். அதாவது தரவு உறுப்புகளுக்கு மதிப்பைக் கொடுக்கவோ அல்லது மதிப்பைப் பெறவோ முடியும். உறுப்பு செயற்கூறுகள் தேவை இல்லை.

Private உறுப்புகள் :

Private அணுகியல்புடன் கூடிய இனக்குழு உறுப்புகளை இனக்குழுவிற்கு வெளியில் இருந்து அணுகமுடியாது. இனக்குழுவின் உறுப்பு செயற்கூறுகள் மட்டுமே தரவு உறுப்புகளை அணுக முடியும். கொடாநிலையாக இனக்குழுவின் உறுப்புகள் அனைத்தும் Private அணுகியல்புடன் அறிவிக்கப்படும்.

Protected உறுப்புகள் :

Protected அணுகியல்புடன் அறிவிக்கப்பட்ட தரவு உறுப்புகளும், உறுப்பு செயற்கூறுகளும் Private அணுகியல்பு கொண்ட உறுப்புகளைப் போலவே செயல்படும். ஆனால் அந்த இனக்குழுவை அடிப்படையாகக் கொண்டு தருவிக்கப்பட்ட இனக்குழுவிற்குள்ளும் அணுக முடியும்.

எடுத்துக்காட்டு

class என்ற சிறப்புச் சொல்லானது இது ஒரு இனக்குழு வரையறை என்பதை நிரல் பெயர்ப்பிக்கு உணர்த்துகிறது

இனக்குழுவின் பெயரானது பயனர் வரையறுக்கும் தரவினத்தின் பெயராகும். இதை பயன்படுத்தி அந்த இனக்குழுவில் பொருள்களை உருவாக்க முடியும்.

```
class student
{
private:
        char name [10];
        int rollno, mark1, mark2, total;
protected:
        void accept();
        void compute();
public:
        void display();
};
```

இவை *private* அணுகியலுடன் கூடிய தரவு உறுப்புகளாகும் அதாவது, இவ்வுருப்புகளை இனக்குழுவிற்கு வெளியிலிருந்து அணுகமுடியாது.

இவை *protected* அணுகியலுடன் கூடிய உறுப்புசெயற்கூறுகளாகும் அதாவது, இவ்வுருப்புகளை இனக்குழுவிற்கு வெளியிலிருந்து அணுகமுடியாது. ஆனால் தருவிக்கப்பட்ட இனக்குழுவிற்குள் அணுகமுடியும்.

இவை *public* அணுகியலுடன் கூடிய உறுப்பு செயற்கூறுகளாகும். இவ்வுருப்புகளை இனக்குழுவிற்கு வெளியிலிருந்து அணுகமுடியும்.

Chapter 14.indd 187 03/09/18 5:49 PM

குறிப்பு

ஓர் இனக்குழுவில் உள்ள அனைத்து உறுப்புகளும் (தரவு உறுப்புகள், உறுப்பு செயல் கூறுகள்) Private அணுகியல்புடன் அறிவிக்கப்பட்டால் அந்த இனக்குழுவானது செயலற்றுப் போகும். இனக்குழுவின் பொருளானது இனக்குழுவின் உறுப்புகள் எதையும் அணுக முடியாது.

செயல்பாடு **1 :**

பின்வரும் குறிமுறையில் பிழைக்கான காரணம் கூறு

(i)	(ii)

class count
{
int first;
int second;
public:
int first;
};
	class item
	{
	int prd;
	};
	item int prdno;

14.1.4 இனக்குழு உறுப்புகளின் வரையறை

இனக்குழுவானது உறுப்புகளை உள்ளடக்கியதாகும். உறுப்புகளானது தரவு உறுப்புகள் மற்றும் உறுப்பு செயற்கூறுகள் என வகைப்படுத்தப்படும். தரவு உறுப்புகள் என்பவை தரவு மாறிகள் எனப்படும். இவை இனக்குழுவின் பண்புக்கூறுகளைக் குறிப்பதாகும். உறுப்பு செயற்கூறுகள் என்பவை ஓர் இனக்குழுவானது குறிப்பிட்ட செயல்பாட்டைச் செய்ய உதவும் செயற்கூறுகளாகும். உறுப்பு செயற்கூறுகளானது வழிமுறைகள் (Methods) எனவும் தரவு உறுப்புகளானது பண்புக்கூறுகள் (attributes) எனவும் அழைக்கப்படும்.

குறிப்பு :

இனக்குழுவானது ஆக்கி மற்றும் அழிப்பி எனப்படும் தனிச்சிறப்பான உறுப்பு செயற்கூறுகளைக் கொண்டிருக்கும்.

எடுத்துக்காட்டு

```
class result
{
private;
        char name [10];
        int rollno,mark1, mark2, total;

public:
        void accept();
        void display();
};
```

தரவு உறுப்புகள்

உறுப்பு செயற்கூறுகள்

14.1.5 உறுப்பு செயற்கூறுகளை வரையறுத்தல் :

இனக்குழுவின் உறுப்புச் செயற்கூறுகளை வரையறுக்காமல் இனக்குழுவின் அறிவிப்பானது முழுமையடையாது. இனக்குழுவின் உறுப்பு செயற்கூறானது இரண்டு விதங்களில் வரையறுக்கப்படுகிறது.

1) இனக்குழுவிற்கு உள்ளே வரையறுத்தல்

2) இனக்குழுவிற்கு வெளியே வரையறுத்தல்

1) இனக்குழுவிற்கு உள்ளே வரையறுத்தல் :

இனக்குழுவின் உள்ளே வரையறுக்கப்படும் செயற்கூறானது inline செயற்கூறு போல இயங்குகின்றன. அவை inline உறுப்பு செயற்கூறுகள் எனவும் அழைக்கப்படுகின்றன.

குறிப்பு :

செயற்கூறானது inline செயல்கூறாக அறிவிக்கப்பட்டால் நிரலின் தொகுப்பு நேரத்தில் அச்செயற்கூறு அழைக்கப்படும் இடங்களில் எல்லாம் நிரல் பெயர்ப்பியானது அச்செயற்கூறின் குறிமுறையை பதிலீடு செய்யும்.

2) இனக்குழுவிற்கு வெளியே வரையறுத்தல் :

இனக்குழுவின் உறுப்பு செயற்கூறானது இனக்குழுவிற்கு வெளியே சாதாரண இனக்குழு வரையறை போலவே வரையறுக்க முடியும்.

Chapter 14.indd 188 03/09/18 5:49 PM

(செயற்கூறு வரையறை பற்றி நீங்கள் நன்கு அறிந்திருந்தால்) இது Outline உறுப்புச் செயற்கூறுகள் (or) non-inline உறுப்புச் செயற்கூறுகள் எனவும் அழைக்கப்படும். உறுப்பு செயற்கூறுகளை இனக்குழுவிற்கு வெளியே வரையறுப்பதற்கு (::) வரையெல்லை செயற்குறியானது (Scope resolution operator) பயன்படுகிறது. உறுப்பு செயற்கூறை இனக்குழுவிற்கு வெளியே வரையறையரை செய்வதற்கு உதவும் பொது வடிவம்.

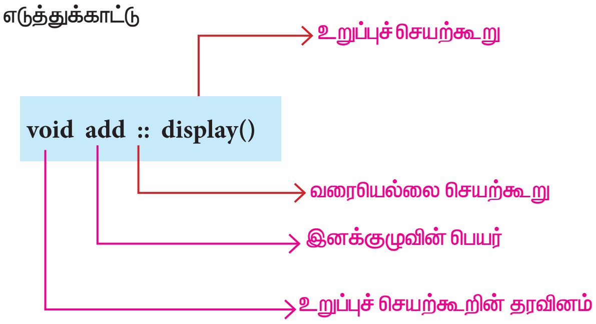
return_type class_name :: function_name (parameter list)

{

செயற்கூறு வரையறை

}

எடுத்துக்காட்டு



விளக்க நிரல் 14.1 Inline மற்றும் Outline உறுப்பு செயற்கூறுகள்

> அணுகியல்பு குறிப்பிடப்படவில்லை எனில் அவை தானமைவாக Private என கருதப்படும்.

```
# include <iostream>
using namespace std;
class Box
{
        double width;          // no access specifier mentioned
public:
        double length;
void printWidth( )             //inline member function definition
        {
                cout<<"\n The width of the box is.."<<width;
        }
void  setWidth( double w );            //prototype of the function
};
void Box :: setWidth(double w)         // outline member function definition
{
width=w;
}
int main( )
{
Box b;  // object for class Box
b.setWidth(10.0);      // Use member function to set the width.
b.printWidth( );               //Use member function to print the width.
return 0;
}
```

வெளியீடு :

The width of the box is .. 10

Chapter 14.indd 189 03/09/18 5:49 PM

குறிப்பு

பல கூற்றுகள், மடக்குகள், switch அல்லது GoTo கூற்றுகள் ஆகியவற்றை கொண்ட உறுப்பு செயற்கூறை Inline செயற்கூறாக அறிவிப்பது ஏற்புடையதல்ல

செயல்பாடு 2 :

```
class area
{
        int s;
public:
        void calc();
};
```

calc () என்னும் செயற்கூறை இனக்குழுவிற்கு வெளியில் வரையறுத்து *(Outline function)* சதுரத்தின் பரப்பளவைக் கண்டறிக.

சுய மதிப்பீடு

1. இனக்குழுவை பொதுவாக மற்றும் C++ சூழலிலும் வரையறு
2. இனக்குழு அணுகியபின் நோக்கம் என்ன?
3. C++ கட்டுரு மற்றும் இனக்குழு ஒப்பிடுக.
4. private மற்றும் public அணுகியல்பை ஒப்பிடுக.
5. Non-inline உறுப்பு செயற்கூறு என்றால் என்ன? பொது வடிவத்தை எழுதுக.

14.2 பொருள்களை உருவாக்குதல்:

இனக்குழு அறிவிப்பு என்பது இனக்குழுவின் பண்புக்கூறுகளை அறிவிப்பதாகும். இனக்குழுவைப் பயன்பாட்டிற்கு கொண்டுவர அந்த இனக்குழுவில் மாறிகளை அறிவிக்க வேண்டும். இனக்குழுவில் அறிவிக்கப்படும் மாறியானது பொருள்கள் (objects) எனப்படும். பொருள்கள் இனக்குழு மாறிகள் எனவும் அழைக்கப்படும்.

(எ.கா)

student s;

மேற்கண்ட கூற்றில் s என்பது student இனக்குழுவின் மாறியாகும்.

இனக்குழுவின் பொருளானது இரண்டு முறைகளில் உருவாக்கப்படுகிறது.

1) முழுதளாவிய பொருள் (Global Object)

2) உள்ளமை பொருள் (Local Object)

1) முழுதளாவிய பொருள் (Global Object)

அனைத்து செயற்கூறுகளின் உடற்பகுதிக்கு வெளியில் அல்லது இனக்குழுவின் அறிவிப்பை முடித்து வைக்கும் கொக்கி வடிவ அடைப்பை (braces) தொடர்ந்து பொருளானது அறிவிக்கப்பட்டால் அது முழுதளாவிய மாறி எனப்படும். இனக்குழுவில்/நிரலில் உள்ள அனைத்து செயற்கூறுகளும் அப்பொருளைப் பயன்படுத்திக் கொள்ள முடியும்.

2) உள்ளமை பொருள்கள் (Local Objects)

ஒரு பொருளானது செயற்கூறினுள் அறிவிக்கப்பட்டால் அது உள்ளமை பொருள் எனப்படும். செயல்கூறுக்கு வெளியில் இருந்து இப்பொருளை அணுக முடியாது.

Chapter 14.indd 190 03/09/18 5:49 PM

14.2 உள்ளமை பொருளின் பயன்பாட்டை விளக்கும் நிரல்

```
# include <iostream>
# include <conio>
using namespace std
class add                //Global class
{
int a,b;
public:
int sum;
void getdata()
{
a=5;
b=10;
sum = a+b;
}
} a1;                    //global object
add a2;                  //global object
int main()
{
add a3;                  //Local object for a global class
a1.getdata();
a2.getdata();
a3.getdata();
cout<<a1.sum;            //public data member accessed from outside the class
cout<<a2.sum;
cout<<a3.sum;
return 0;
}
```

முழுதளாவிய பொருளை முழுதளாவிய இனக்குழுவிற்கு மட்டுமே அறிவிக்க முடியும். இனக்குழுவின் வரைறையானது அனைத்து செயற்கூறுகளின் உடற்பகுதிக்கு வெளியே குறிப்பிடப்பட்டால் அது முழுதளாவிய இனக்குழுவாகும்.

வெளியீடு :

```
151515
```

செயல்பாடு 3

பின்வரும் நிரல் குறிமுறையில் உள்ள பிழைகளை கண்டறிக

```
class A
{
        float x;
        void init()
        {
        A a1;
        X1=1.5;
        }};
void main()
{    A1.init(); }
```


Chapter 14.indd 191   03/09/18 5:49 PM

14.3 பொருள்களுக்கான நினைவக இட ஒதுக்கீடு

இனக்குழு வரையறுப்பின் ஒரு பகுதியாக உறுப்பு செயற்கூறுகள் வரையறுக்கப்பட்டால் மட்டுமே அவை உருவாக்கப்பட்டு நினைவகத்தில் இருத்தப்படுகின்றன. அந்த இனக்குழுவைச் சார்ந்த அனைத்துப் பொருள்களும் ஒரே உறுப்பு செயற்கூறை பயன்படுத்திக் கொள்வதால், ஒவ்வொரு பொருள் உருவாக்கப்படும் போதும் உறுப்பு செயற்கூறுகளுக்கு தனித்தனி நினைவக இடம் ஒதுக்கப்படுவதில்லை. உறுப்பு மாறிகளுக்குத் தேவையான நினைவக இடம் மட்டும் ஒவ்வொரு பொருள் உருவாக்கப்படும் போதும் தனித்தனியே ஒதுக்கப்படும். ஏனெனில் ஒவ்வொரு பொருளின் தரவு உறுப்புகள் / உறுப்பு மாறிகள் வெவ்வேறு மதிப்புகளைக் கொண்டிருக்கும்.

கீழ்காணும் இனக்குழு அறிவிப்பு நிரலை 14.3 நோக்குக.

```
# include <iostream>
using namespace std;
class product
{
        int code, quantity;
        float price;
        public:
        void assignData();
        void Print();
};
int main()
{
        product p1, p2;
        cout<<"\n Memory allocation for object p1 " <<sizeof(p1);
        cout<<"\n Memory allocation for object p2 " <<sizeof(p2);
        return 0;
}
```

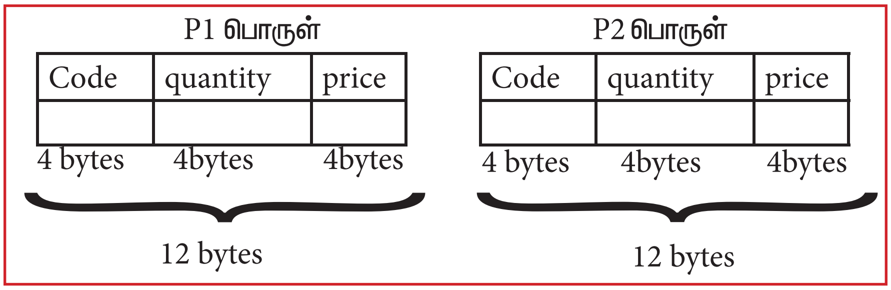
> இனக்குழுவில் பொருளானது உருவாக்கப்பட்ட பின்னர் தான் அதன் உறுப்புகளுக்கு நினைவக இடமானது ஒதுக்கப்படும்.

வெளியீடு :

```
        Memory allocation for object p1  12
        Memory allocation for object p2  12
```

நிரலில் உள்ள assign data () மற்றும் print () உறுப்பு செயற்கூறுகளுக்குப் பொதுவான நினைவகப் பகுதி ஒதுக்கீடு செய்யப்படும். P1 மற்றும் P2 ஆகிய இரண்டு பொருள்களுக்கான தரவு உறுப்புகள் அனைத்தும் பொதுவானப் பகுதியில் உள்ள உறுப்பு செயற்கூறுகளை அணுக முடியும்.

P1 மற்றும் P2 பொருள்களுக்கான நினைவக ஒதுக்கீடு விளக்கம்



Chapter 14.indd 192
03/09/18 5:49 PM

செயல்பாடு 4

கீழ்கண்ட குறிமுறையில் S1 மற்றும் S2 பொருள்களுக்கான நினைவக அளவு யாது?

```
class sum
{
        int n1,n2;
        public:
        void add(){int n3=10;n1=n2=10;}
}  s1,s2;
```

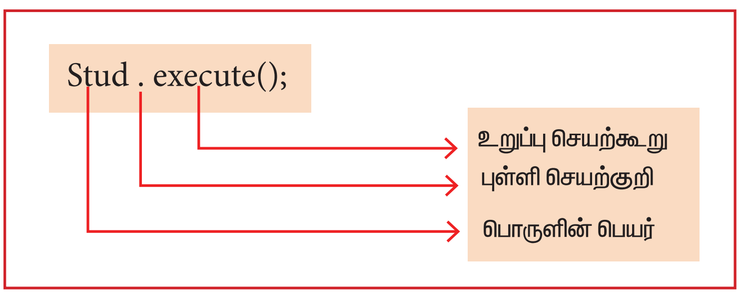
14.4 இனக்குழு உறுப்புகளை அணுகுதல்

இனக்குழுவின் உறுப்புகளை அணுகுவதற்கு புள்ளி (. DOT operator) செயற்குறியானது பயன்படுகிறது. இனக்குழு பொருளின் பெயரைத் தொடர்ந்து புள்ளி செயற்குறியும், அதைத் தொடர்ந்து அணுக வேண்டிய தரவு உறுப்பு / உறுப்பு செயற்கூறின் பெயரைத் தர வேண்டும்.

இனக்குழுவின் உறுப்புகளை அணுகப் பயன்படும் புள்ளி செயற்குறியின் தொடரியல்:

பொருளின் பெயர் . உறுப்பு செயற்கூறின் பெயர் (மெய்யான அளபுருக்கள்) ;

(எடுத்துக்காட்டு)



விளக்க நிரல் 14.4 பொருளின் பயன்பாட்டை விளக்கும் நிரல்

```
#include<iostream>
using namespace std;
class compute
{
        int n1,n2; //private by default
public :
        int n;
        int add (int a, int b)    //inline member function
        {
                int c=a+b;        //int c ; local variable for this function
                return c;
        }
```

பொருளின் உறுப்புச் செயற்கூறை அழைத்தல் என்பது பொருளுக்கு செய்தியை அனுப்புவது அல்லது தொடர்பு கொள்ளுதல் ஆகும்.

Chapter 14.indd 193 03/09/18 5:49 PM

```
        int prd (int a, int b)       //inline member function
        {
                int c=a*b;
                return c;
        }
};      // end of class specification
compute c1,c2;          //global object
int main()
{
        c1.n =c1.add(12,15);  //member function is called
        c2.n =c2.add(8,4);
        cout<<"\n Sum of object 1  "<<c1.n;
        cout<<"\n Sum of object 2  "<<c2.n;
        cout<<"\n Sum of the two objects are  "<<c1.n+c2.n;
        c1.n=c1.prd(5,4);
        c2.n=c2.prd(2,5);
        cout<<"\n Product of object 1  "<<c1.n;
        cout<<"\n Product of object 2  "<<c2.n;
        cout<<"\n Product of the two objects are  "<<c1.n*c2.n;
        return 0;
}
```

வெளியீடு :

```
        Sum of object 1  27
        Sum of object 2  12
        Sum of the two objects are  39
        Product of object 1  20
        Product of object 2  10
        Product of the two objects are  200
```

செயல்பாடு 5

1) மதிப்பைத் திரும்ப அனுப்பாத display () – எனும் உறுப்பு செயற்கூறை வரையறுக்கவும். அச்செயற்கூறானது இனக்குழுவின் தரவு உறுப்புகளின் மதிப்புகளை வெளிப்படுத்த வேண்டும்.
2) மேற்கண்ட நிரல் குறிமுறையில் தேவையான மாற்றங்களைச் செய்து வெளியீட்டைக் காணவும்.

14.5 பொருள்களின் அணிகள்

இனக்குழு உறுப்புகளை (பொருள்களை) வரிசையாகக் கொண்ட அணிகள் பொருள்களின் அணிகள் எனப்படும். மற்ற வகையான அணிகளை அறிவித்து வரையறுப்பது போன்றே பொருள்களின் அணிகளையும் உருவாக்கலாம்.

Chapter 14.indd 194 03/09/18 5:49 PM

14.5 பொருள்களின் அணிகளைப் பயன்படுத்தி C++ நிரல்

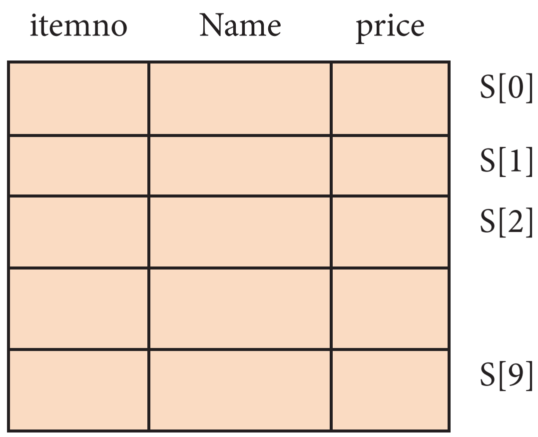
> gets () செயற்கூறானது இடைவெளிகளுடன் கூடிய தரவுகளை உள்ளீடாக பெறும். cin முறையில் அவ்வாறு செய்ய இயலாது.

```
#include<iostream>
#include<stdio.h>
using namespace std;
class stock
 {
        int itemno;
        char itemname[10];      // array as member variable
        float price;
public:
        void getdata()
        {
        cin>>itemno>>price;
        gets(itemname);
}
void putdata()
{
        cout<<'\n'<<itemno<<'\t'<<itemname<<'\n'<<price;
}
};
int main()
{
stock s[10];
int i;
cout<<"Enter the details\n";
        for( i=0;i<10;i++)
                s[i].getdata();
cout<<"\n Item Details\n";
cout<<"\nITEM NO \t ITEM NAME\t PRICE"<<endl;
        for( i=0;i<10;i++)
        for( i=0;i<10;i++)
                s[i].putdata();
return 0;
}
```

வெளியீடு :

```
        Item Details
        ITEM NO     ITEM NAME                   PRICE
        101         APSARA PENCIL               Price10.5
        102         FABER CASTELL PENCIL        Price12.0001
        103         NATARAJ PENCIL              Price9.75
        201         PILOT V7 PEN                Price50
        201         CELLO BUTTERFLOW            Price10
        202         PARKER PEN                  Price450
        202         PILOT FOUNTAIN PEN          Price600
        301         APSARA ERASER               Price5
        302         FABER CASTELL ERASER        Price15.0001
        303         NATARAJ ERASER              Price3
```


Chapter 14.indd 195 03/09/18 5:49 PM



10 இனக்குழு பொருள்களைக் கொண்ட அணிக்கான நினைவக ஒதுக்கீடு.

மேற்கண்ட நிரலானது 10 பொருள்களின் விபரங்களைப் பெற்று அவற்றை வெளியீடாகத் தருகின்றது. மேற்கண்ட நிரலில் 10 இனக்குழு பொருள்கள் உருவாக்கப்பட்டிருப்பதையும், உறுப்பு செயற்கூறுகள் மடக்குகளின் உதவியுடன் அவற்றை அணுகுவதையும் காண்க.

குறிப்பிட்ட பொருளுக்கான putdata () செயற்கூற்றைச் செயல்படுத்துவதற்கு (எ.கா) 6 வது பொருளிற்கான செயற்கூற்றை அணுகும் முறை கீழேக் கொடுக்கப்பட்டுள்ளது.

S[5].putdata ();

அணியின் கீழொட்டானது 0 ல் இருந்து துவங்கும். எனவே 5 வது கீழொட்டானது 6 வது பொருளினைக் குறிக்கும்.

14.6 இனக்குழுவின் செயற்கூறுகள்

இனக்குழு தரவு உறுப்புகளின் மீது செய்யப்படும் செயல்பாடுகள் அனைத்தும் உறுப்பு செயற்கூறுகளைப் பயன்படுத்திச் செய்யப்படுகின்றன. இவ்வகையான செயல்பாடுகள் அனைத்தும் உறுப்பு செயல்கூறுகளுக்குள் வரையறுக்கப்பட்டிருக்கும்.

14.6.1 inline மற்றும் Outline உறுப்பு செயற்கூறுகளைப் பொருளானது அணுகுதல் :

இனக்குழு உறுப்பு செயற்கூறுகளின் வரையறையானது இனக்குழு அறிவிப்பிற்கு உள்ளேயே அல்லது வெளியிலோ வரையறுக்கப்படலாம்.

விளக்க நிரல் :

பின்வரும் பண்புகளுடன் கூடிய sales என்னும் இனக்குழுவை உருவாக்கவும்.

private members of class salesman	
sno	integer(salesperson number)
sname	15 characters
hrwrk ,wage	float (hours worked and wage per hour)
totwage	float(*hrwrk* * *wage*)
calcwg( )	a function to find *hrwrk* * *wage* and store it in *totwage*

Chapter 14.indd 196 03/09/18 5:49 PM

public members of class salesman	
in_data( )	a function to accept values for *sno*, *sname*, *hrwrk*, *wage* and invoke calcwg( ) to calculate *totwage*
out_data( )	a function to display all the data members on the screen. you should give definitions of functions as outline

நிரல் 14.6 Inline மற்றும் Outline செயற்கூறுகள் பற்றிய விளக்க நிரல்

> Inline உறுப்பு செயற்கூறின் வரையறையானது அதை அழைக்கும் அனைத்து செயற்கூறுகளுக்கும் மேலாக இடம் பெறல் வேண்டும்.

> Outline உறுப்பு செயற்கூறின் முன்வடிவானது இனக்குழுவின் அறிவிப்புக்குள் கொடுக்கப்படுதல் வேண்டும். முன்வடிவானது உறுப்பு செயற்கூறுகளின் அணுகியல்புகளை நிரல் பெயர்ப்பிக்கு உணர்த்தும்.

```
#include<iostream>
using namespace std
class sales
{
        int sno;
        char sname [15];
        float hrwrk,wage,totwage;
void calcwg( )
        {
                totwage=hrwrk * wage;
        }
public :
        void in_data();
        void out_data();
};
void sales :: in_data()
{
        wage=75.26;
        totwage=0.0;
        cout<<"\nEnter the salesperson id";cin>>sno;
        cout<<"\nEnter the name" ;gets(sname);
        cout<<"\nEnter the hours worked" ;cin>>hrwrk;
        calcwg();   //member function called inside by another member function
}
void sales :: out_data()
{
        cout<<"\n Wage slip ";
        cout<<"\n ~~~~~~~~ ";
        cout<<"\nID : " << sno;
        cout<<"\nName : " <<sname ;
        cout<<"\nHours worked :" <<hrwrk;
        cout<<"\nTotal Wage :"<<setprecision(2)<<totwage;
}
```

```
int main()
{
	sales sal;
	sal.in_data();
	sal.out_data();
	return 0;
}
```

வெளியீடு :

```
	Enter the salesperson id 1201
	Enter the name ARUL
	Enter the hours worked 7
	 Wage slip
	            ~~~~~~~~~
	ID : 1201
	Name : ARUL
	Hours worked :7
	Total Wage :526.82
```

14.7 உறுப்பு செயற்கூறுகளின் பின்னலமைப்பு:

Public அணுகியல்புகளைக் கொண்ட தரவு உறுப்புகள், உறுப்பு செயற்கூறுகளை மட்டுமே இனக்குழுவின் பொருளானது புள்ளி செயற்குறி மூலம் அணுக முடியும் என்பது நாம் அறிந்ததே ஆகும். ஓர் இனக்குழுவில் உள்ள உறுப்பு செயற்கூறானது அதே இனக்குழுவில் உள்ள மற்றொரு உறுப்பு செயற்கூறைப் புள்ளி செயற்குறி இல்லாமல் அழைக்க முடியும். இது உறுப்பு செயற்கூறுகளின் பின்னலமைப்பு எனப்படும்.

விளக்க நிரல் 14.7 உறுப்பு செயற்கூறுகளின் பின்னலமைப்பு பற்றி விளக்கும் நிரல்

```
#include<iostream>
using namespace std
class nest
{
	int a;
	int square_num( )
	{
		return a* a;
	}
	public:
	void input_num( )
	{
		cout<<"\nEnter a number ";
		cin>>a;
	}
```

ஒரு உறுப்பு செயற்கூறானது அதே இனக்குழுவில் உள்ள மற்றொரு உறுப்பு செயற்கூற்றை அணுக பொருளானது தேவையில்லை.

Chapter 14.indd 198 03/09/18 5:49 PM

```
            int cube_num( )
            {
                        return a* a*a;
            }
void disp_num()
{
            int sq=square_num();            //nesting of member function
            int cu=cube_num();              //nesting of member function
            cout<<"\nThe square of "<<a<<" is   " <<sq;
            cout<<"\nThe cube of "<<a<<" is   " <<cu;
}
};
int main()
 {
            nest n1;
            n1.input_num();
            n1.disp_num();
            return 0;
}
```

வெளியீடு :

```
            Enter a number 5
            The square of 5 is 25
            The cube of 5 is 125
```

மேற்கண்ட நிரலில் square_num () மற்றும் cube_num () என்ற இரண்டு உறுப்பு செயற்கூறுகள் disp_num () எனும் உறுப்பு செயற்கூறால் அழைக்கப்பட்டுள்ளன. இரண்டு உறுப்பு செயற்கூறுகளும் வெவ்வேறு வகையான அணுகியல்புடன் வரையறுக்கப்பட்டுள்ளன.

குறிப்பு

> இனக்குழுவின் உறுப்பு செயற்கூறானது Public அணுகியல்புடன் அறிவிக்கப்பட்ட உறுப்பு செயற்கூறுகளைக் கொண்டு அணுகுவதுடன், அதே இனக்குழுவில் Private அணுகியல்புடன் வரையறுக்கப்பட்டுள்ள உறுப்பு செயற்கூறுகளைக் கொண்டும் அணுக முடியும்.

14.7.1 வரையெல்லை செயற்குறி (scope resolution operator)

ஒரே பெயரினைக் கொண்ட ஒன்றுக்கும் மேற்பட்ட மாறிகள் வெவ்வேறு தொகுதியில் அறிவிக்கப்பட்டிருந்தால், அவற்றில் முழுதளாவிய மாறிகளைக் குறிப்பிட (::) வரையெல்லை செயற்குறியானது பயன்படுகிறது.

Chapter 14.indd 199 03/09/18 5:49 PM

நிரல் 14.8 வரையெல்லை செயற்குறியின் பயன்பாட்டை விளக்கும் நிரல்

(::) என்ற செயற்குறி உறுப்பு செயற்கூறின் இனக்குழுவை அடையாளம் காணப் பயன்படுகிறது.

```
#include<iostream>
using namespace std;
int a=100;
class A
{
        int a;
        public:
void fun()
        {
        a=20;
        a+=::a;         //using global variable value
        cout<<a;
        } };
int main()
{
        clrscr();
        A a1;
        a1.fun();
        retun 0;
}
```

வெளியீடு :

```
        120
```

மேற்கண்ட நிரலில் இனக்குழுவின் தரவு உறுப்பும், முழுதளாவிய மாறியும் ஒரே பெயரினைக் கொண்டுள்ளன. முழுதளாவிய மாறியைக் குறிக்க (விரிவான வரையெல்லை கொண்ட) (::) வரையெல்லை செயற்குறியானது பயன்படுகிறது.

14.8 செயற்கூறுகளுக்குப் பொருளை அளபுருக்களாக அனுப்புதல்

இனக்குழுவின் உறுப்பு செயற்கூறுகளுக்கு அளபுருவாக மற்ற தரவுகளை / தரவு வகைகளை அனுப்புவது போல் பொருளையும் அளபுருவாக அனுப்பி வைக்க முடியும். பொருளானது இரண்டு விதங்களில் அளபுருவாக அனுப்பி வைக்க முடியும்.

1) மதிப்பு மூலம் அனுப்புதல் (Call by Value)

2) குறிப்பு மூலம் அனுப்புதல் (Call by Reference)

14.8.1 மதிப்பு மூலம் அனுப்புதல்

இனக்குழுவின் உறுப்பு செயற்கூறுகளுக்கு மதிப்பு மூலம் அனுப்புதல் முறையில் பொருளை அளபுருவாக அனுப்பும்போது, செயற்கூறானது பொருளினை நகல் எடுத்து அதன் மீது செயல்பாடுகளைச் செயல்படுத்தும்.

எனவே செயற்கூறினுள் பொருள்களின் மீது செய்யப்படும் எந்த மாற்றமும் (முறையான அளபுருவாக உள்ள பொருளின் தரவு உறுப்புகளின் மீது) மெய்யான பொருளின் (மெய்யான அளபுருவாக உள்ள பொருளின் தரவு உறுப்புகளின் மீது) மீது எந்த பாதிப்பையும் ஏற்படுத்தாது.

விளக்க நிரல் : 14.9 மதிப்பு மூலம் அனுப்புதல் முறையானது செயல்படும் விதம் பற்றிய நிரல்

```
#include <iostream>
using namespace std;
class Sample
{
private:
        int num;
public:
        void set (int x)
        {
        num = x;
        }
void pass(Sample obj1, Sample obj2)  //objects are passed
{
        obj1.num=100;    // value of the object is changed inside the function
        obj2.num=200;    // value of the object is changed inside the function
        cout<<"\n\n Changed value of object1 "<<obj1.num;
        cout<<"\n\n Changed value of object2 "<<obj2.num;
}
}
        void print( )
        {
                cout<<num;
        }
};
int main()
{
        //object declarations
        Sample s1;
        Sample s2;
        Sample s3;
        //assigning values to the data member of objects
        s1.set(10);
        s2.set(20);
        cout<<"\n\t\t Example program for pass by value\n\n\n";
        //printing the values before passing the object
        cout<<"\n\nValue of object1 before passing";
        s1.print();
        cout<<"\n\nValue of object2 before passing ";
         s2.print();
        //passing object s1 and s2
         s3.pass(s1,s2);
         //printing the values after returning to main
          cout<<"\n\nValue of object1 after passing ";
```

ஒரு பொருளுக்கு மற்றொரு பொருளை கட்டுருவின் பொருளைப் போல மதிப்பிருத்தலாம்.

Chapter 14.indd 201 03/09/18 5:49 PM

```
        s1.print();
        cout<<"\n\nValue of object2 after passing ";
        s2.print();
       return 0;
}
```

வெளியீடு :

```
Example program for PASS BY VALUE
Value of  object1 before passing 10
Value of object2 before passing 20
Changed value of object 1 100
Changed value of object 200
Value of object 1 after passing 10
Value of object 2 after passing 20
```

மேற்கண்ட நிரலில் s1 மற்றும் s2 பொருளானது pass () செயல்கூறுக்கு அளபுருக்களாக அனுப்பி வைக்கப்பட்டுள்ளது. s1 மற்றும் s2 பொருளானது முறையே Obj1 மற்றும் Obj2 என்ற முறையான அளபுருக்களில் நகல் எடுக்கப்படும். முறையான அளபுருக்களில் உள்ள num என்ற தரவு உறுப்பின் மதிப்பானது செயற்கூறினுள் மாற்றப்பட்டுள்ளது. ஆனால் செயற்கூறினுள் செய்யப்படும் மாற்றமானது s1 மற்றும் s2 பொருளின் தரவு உறுப்புகளில் மாற்றத்தை ஏற்படுத்தாது.

14.8.2 குறிப்பு மூலம் அனுப்புதல்

உறுப்பு செயற்கூறுகளுக்கு குறிப்பு மூலம் பொருளை அளபுருவாக அனுப்பும் போது, பொருளின் நினைவக முகவரியானது அளபுருவாக அனுப்பி வைக்கப்படும். எனவே அழைக்கப்படும் செயற்கூறானது மெய்யான அளபுருவாக உள்ள பொருளின் மீதே செயல்பாட்டைச் செயல்படுத்தும். செயற்கூறில் (முறையான அளபுருக்களின் மீது) செய்யப்படும் எந்த மாற்றமும் மெய்யான அளபுருவாக உள்ள பொருளின் மீதும் பாதிப்பை ஏற்படுத்தும்.

விளக்க நிரல் 14.10 குறிப்பு மூலம் அனுப்புதல் முறை செயல்படும் விதம் பற்றிய நிரல்

```
#include <iostream>
using namespace std;
class Sample
{
private:
        int num;
public:
        void set (int x)
        {
                num = x;
        }
```


Chapter 14.indd 202 03/09/18 5:49 PM

```
void pass(Sample &obj1, Sample &obj2)  //objects are passed
{
        obj1.num=100;    // value of the object is changed inside the function
        obj2.num=200;    // value of the object is changed inside the function
        cout<<"\n\n Changed value of object1 "<<obj1.num;
        cout<<"\n\n Changed value of object2 "<<obj2.num;
}
        void print()
        {
                cout<<num;
        }
};
nt main()
{
        clrscr();
        //object declarations
        Sample s1;
        Sample s2;
        Sample s3;
        //assigning values to the data member of objects
        s1.set(10);
        s2.set(20);
        cout<<"\n\t\t Example program for pass by reference\n\n\n";
        //printing the values before passing the object
        cout<<"\n\nValue of object1 before passing";
        s1.print();
        cout<<"\n\nValue of object2 before passing ";
         s2.print();
        //passing object s1 and s2
         s3.pass(s1,s2);
         //printing the values after returning to main
        cout<<"\n\nValue of object1 after passing ";
         s1.print();
         cout<<"\n\nValue of object2 after passing ";
         s2.print();
        return 0;
}
```

வெளியீடு :

```
Example program for PASS BY REFERENCE
Value of object1 before passing10
Value of object 2 before passing 20
Changed value of object1 100
Changed value of object2 200
Value of object1 after passing 100
Value of object2 after passing 200
```

மேற்கண்ட நிரலில் S1 மற்றும் S2 பொருளானது குறிப்பு மூலம் அனுப்புதல் முறையில் pass () என்ற உறுப்பு செயற்கூறுக்கு அளபுருக்களாக அனுப்பப்பட்டுள்ளன. ஆகையால் Obj1 மற்றும் Obj2 அளபுருக்களானது முறையே S1 மற்றும் S2 அளபுருக்களின் மாற்றுப் பெயர்களாகச் செயல்படுகின்றன. செயற்கூறினுள் num என்ற தரவு உறுப்பின் மீது செய்யப்படும் மாற்றமானது S1 மற்றும் S2 பொருள்களின் தரவு உறுப்பின் மீது பாதிப்பை ஏற்படுத்தும்.

14.9 செயற்கூறு பொருளைத் திரும்ப அனுப்புதல்

உறுப்பு செயற்கூறுகளானவை அளபுருக்களாக பொருளைப் பெறுவதோடு மட்டுமல்லாமல், செயல்பட்டு முடிந்த பின்பு அவை மதிப்பாக ஒரு பொருளைத் திரும்ப அனுப்பக் கூடியவை ஆகும்.

விளக்க நிரல் 14.11 அழைக்கப்படும் செயற்கூறு பொருளை மதிப்பாக திரும்ப அனுப்புதல்

```
#include <iostream>
using namespace std;
class Sample
{
    private:
        int num;
    public:
    void set (int x)
    {
    num = x;
    }
    Sample pass(Sample obj1, Sample obj2)  //
    {
      Sample s4;
      s4.num=obj1.num+obj2.num;
      return s4;
    }
    }
```


Chapter 14.indd 204 03/09/18 5:49 PM

```
            void print()
            {
            cout<<num;
            }
};
int main()
{
/object declarations
            Sample s1;
            Sample s2;
            Sample s3;
            //assigning values to the data member of objects
            s1.set(10);
            s2.set(20);
            cout<<"\n\t\t Example program for Returning an object \n\n\n";
            //passing object s1 and s2
            s3=s3.pass(s1,s2);
            //printing the values of object
            cout<<"\nThe value of s1.num is ";
            s1.print();
            cout<<"\nThe value of s2.num is ";
            s2.print();
            //printing the sum
            cout<<"\nThe sum s3.num is  ";s3.print();
            return 0;
}
```

வெளியீடு :

```
            Example program for Returning an object
            The value of s1.num is 10
            The value of s2.num is 20
            The sum s3.num is  30
```

14.10 பின்னலான இனக்குழு (Nested class)

ஓர் இனக்குழு ஆனது மற்றொரு இனக்குழுவின் உறுப்பாக (ஓர் இனக்குழுவிற்குள் மற்றொரு இனக்குழு அமைந்திருந்தால்) அமைந்திருந்தால் அது பின்னலான இனக்குழு எனப்படும். இனக்குழுக்களுக்கு இடையே நிலவும் உறவு முறையானது கண்டெய்னர் ஷிப் (container ship) எனப்படும்.

பின்னலான இனக்குழுவை இரண்டு வழிகளில் உருவாக்கலாம்.

1) ஓர் இனக்குழுவிற்குள் மற்றொரு இனக்குழுவை அமைத்தல்

2) ஓர் இனக்குழுவின் பொருளை மற்றொரு இனக்குழுவின் உறுப்பாக அறிவித்தல்.

ஓர் இனக்குழுவினுள் மற்றொரு இனக்குழுவை வரையறுத்தல் :

ஓர் இனக்குழுவானது மற்றொரு இனக்குழுவினுள் அறிவிக்கப்பட்டிருந்தால், உள்ளமைக்கப்பட்ட இனக்குழுவானது உள்ளமை இனக்குழு (Nested Class) எனவும் வெளியே உள்ள இனக்குழுவானது அடைப்பு இனக்குழு (Enclosing Class) எனப்படும். அடைப்பு இனக்குழுவின் (Enclosing Class) Public அணுகியல்பு பகுதியில் உள்ளமை இனக்குழு ஆனது Private அணுகியல்புடன் வரையறுக்கப்படலாம்.

விளக்க நிரல் : 14.12 பின்னலான இனக்குழுவை விளக்கும் நிரல்

```
#include<iostream>
using namespace std;
class enclose
{
private:
        int x;
        class nest
        {
private :
        int y;
public:
int z;
void prn()
{
        y=3;z=2;
        cout<<"\n The product of"<<y<<'*'<<z<<"= "<<y*z<<"\n";
}
}; //inner class definition over
nest n1;
public:
nest n2;
void square()
{
n2.prn(); //inner class member function is called by its object
        x=2;
        n2.z=4;
        cout<<"\n The product of " <<n2.z<<'*'<<n2.z<<"= "<<n2.z*n2.z<<"\n";
        cout<<"\n The product of " <<x<<'*'<<x<<"= "<<x*x;
        }
};              //outer class definition over
int main()
{
enclose e;
        e.square();   //outer class member function is called
}
```

வெளியீடு :

```
The product of 3*2=6
The product of 4*4=16
The product of 2*2=4
```

மேற்கண்ட நிரலில் enclose என்ற இனக்குழுவினுள் nest – என்ற இனக்குழு ஆனது வரையறுக்கப்பட்டுள்ளது. nest – இனக்குழுவில் உருவாக்கப்படும் பொருளின் மூலமாக enclose – இனக்குழு nest – இனக்குழுவை அணுக முடியும்.

14.10.1 கன்டைனர் ஷிப்- C++

ஓர் இனக்குழுவின் பொருளானது மற்றொரு இனக்குழுவின் உறுப்பாக அறிவிக்கப்படும் முறையானது கன்டைனர் இனக்குழு எனப்படும். இம்முறையில் ஒரு இனக்குழுவின் பொருளானது மற்றொரு இனக்குழுவினுள் அறிவிக்கப்பட்டிருக்கும்.

விளக்க நிரல் 14.13 containership இனக்குழுவுடன் கூடிய நிரல்

```
#include<iostream>
using namespace std;
class outer
{
        int data;
        public:
        void get();
};
class inner
{
        int value;
        outer ot;        // object ot of class outer is declared in class inner
        public:
        void getdata();
};
void outer :: get()
{
        cout<<"\nEnter a value";
        cin>>data;
        cout<<"\nThe given value is "<<data;
}
void inner :: getdata()
{
        cout<<"\nEnter a value";
        cin>>value;
        cout<<"\nThe given value is "<<value;
        ot.get();    //calling of get() of class outer in getdata() of class inner
}
int main()
{
        inner in;
        in.getdata();
        return 0;
}
```


Chapter 14.indd 207 03/09/18 5:49 PM

```
int main()
{
        inner in;
        in.getdata();
        return 0;
}
```

வெளியீடு :

```
        Enter a value10
        The given value is 10
        Enter a value 20
        The given value is 20
```

மேற்கண்ட நிரலில் வெளி மற்றும் உள் இனக்குழுக்கள் தனித்தனியாக வரையறுக்கப்பட்டுள்ளன. ஆனால் உள் இனக்குழுவின் உறுப்பான ot என்ற பொருள் மூலம் இரண்டு இனக்குழுக்களும் இணைக்கப்பட்டுள்ளன.

14.11 ஆக்கிகள் – ஓர் அறிமுகம்

இனக்குழுவை வரையறுப்பதன் மூலம் ஒரு புதிய பயனர் வரையறுக்கும் தரவினத்தை உருவாக்க முடியும். இனக்குழுவின் சான்றுருவை சான்றாக்கல் (உருவாக்கி, தொடங்கி வைத்தல்) அவசியமாகும். ஆக்கிகளைப் பயன்படுத்தி பொருள்களைச் சான்றாக்குதல் முடியும்.

14.11.1 ஆக்கிகளின் தேவை

C++ ல் உள்ள அணி அல்லது கட்டுரு, அதன் அறிவிக்கும் நேரத்தில் தொடங்கி வைக்கப்படுகிறது.

எடுத்துக்காட்டாக

```
struct sum
   {
        int n1,n2;
   };
class add
  {
        int num1,num2;
  };
int  main()
   {
        int arr[]={1,2,3};        //declaration and initialization of array
        sum s1={1,1};            //declaration and initialization of structure object
        add  a1={0,0};          // class object declaration and initialization throws
                                                                   compilation error
    }
```

இனக்குழு உறுப்பு செயற்கூறானது அதனோடு தொடர்புடைய அணுகியல்பு எதுவாக இருந்தாலும் அணுக முடியும்.

இனக்குழு பொருள் அதன் அறிவிப்பு நேரத்தில் தொடங்கி வைப்பது (initialization) கட்டுரு அல்லது அணியை தொடங்குதல் போன்றது அல்ல. ஏனென்றால், இனக்குழு உறுப்புகள் அதனோடு தொடர்புடைய அணுகியல்பைக் கொண்டிருக்கும் (public, private or protected). அதனால், இனக்குழுக்கள் ஆக்கிகள்

என்னும் சிறப்பு உறுப்பு செயற்கூறுகளைக் கொண்டிருக்கும். ஆக்கி செயற்கூறு இனக்குழு பொருளைத் தொடங்கி வைக்கிறது.

14.12 அறிவிப்பு மற்றும் வரையறுப்பு

ஓர் இனக்குழுவின் சான்றுரு பயன்பாட்டுக்கு வரும்போது ஆக்கி எனப்படும் சிறப்பு செயற்கூறு இயக்கப்படுகிறது. ஆக்கியின் பெயர் இனக்குழுவின் பெயராகவே இருக்க வேண்டும். இது எதையும் திருப்பி அனுப்பாது. இதற்கு எந்த தரவினத்தோடும் தொடர்பு இல்லை.

இனக்குழு வரையறுப்பின் உள்ளே அல்லது வெளியே இதனை வரையறுக்கலாம்.

எடுத்துக்காட்டு : 1

பின்வரும் நிரல் 14.14 ஆக்கியை இனக்குழுவின் உள்ளே வரையறுப்பதை விளக்குகிறது.

```
#include<iostream>
using namespace std;
class Sample
        {
int i,j;
        public :
int k;
        Sample()
        {
                i=j=k=0;//constructor defined inside the class
        }
        };
int main()
{
Samples1;
    return 0;
}
```

வெளியீடு :

செயல்பாடு 6

மேற்கண்ட நிரலுக்கு வெளியீடு இல்லாததன் காரணத்தை விவரி.

Chapter 14.indd 209 03/09/18 5:49 PM

எடுத்துக்காட்டு : 2

பின்வரும் நிரல் 14.15 ஆக்கியை இனக்குழுவின் வெளியே வரையறுப்பதை விளக்குகிறது.

```
#include<iostream>
using namespace std;
class Data
        {
int p,q;
            public :
int r;
              Data();    //only prototype to be specified here to intimate its access
                                                                         specifier
       };
Data ::Data()
{
       p=q=r=0;          // constructor defined outside the class
 }
int main()
{
     Data d1;
return 0;
}
```

குறிப்பு

ஆக்கிகளை இனக்குழுவின் private அல்லது public பகுதிகளில் வரையறுத்துக் கொள்ளலாம். ஆனால் ஆக்கிகளை public பகுதியில் வறையறுக்க பரிந்துரைக்கப் படுகிறது. ஏனென்றால் எந்த செயற்கூறிலும் பொருளை உருவாக்க முடியும்.

Chapter 14.indd 210 03/09/18 5:49 PM

14.16 private அணுகியல்புக்கு உள்ளே ஆக்கிகளை வரையறுத்தலை நிரல்.

```
#include<iostream>
using namespace std;
class X
 {
int num;
int num;
X()
     {
```

```
num=k=0;
     }
  public:
int k;
 };
int main()
{
X x;   // The constructor of X cannot accessed by main() as main() is a
            //non member function
                       //and the compiler throws error message        [Error]    'X::X()'
is                    //private
return 0;
}
```

14.12.1 ஆக்கியின் செயல்பாடுகள் :

இனக்குழுவின் சான்றுரு உருவாக்கப்படும் போது அல்லது அறிவிப்பின் போது, சிறப்பு தொடங்கு உறுப்பு செயற்கூறான ஆக்கி தானாகவே தொடங்கி இயக்கப்படுகிறது. ஆக்கியின் முக்கிய செயல்பாடுகள் ஆவன :

1. பொருளுக்கு நினைவகத்தில் இடம் ஒதுக்குகிறது
2. இனக்குழுப் பொருளின் உறுப்புகளில் தொடக்க மதிப்பு இருத்துகிறது.

இனக்குழுப் பொருளைத் தொடங்கி வைப்பதற்கான மாற்றுவழி உள்ளது. ஆனால் இந்த வழியில் உறுப்பு செயற்கூறினை வெளியில் இருந்து அழைக்க வேண்டும்.

Chapter 14.indd 211 03/09/18 5:49 PM

பின்வரும் நிரல் 14.17 உறுப்பு செயற்கூறு தரவு உறுப்பைத் தொடங்கி வைத்தலை விளக்குகிறது.

```
#include<iostream>
using namespace std;
class Sample
 {
int i, j;
      public :
int k;
void getvalue()
      {
           i=j=k=0;                //member function
       }
       };
int main()
{
Sample s1;
     s1.getvalue();              //member function initializes the class object
return 0;
}
```

பொருள் உருவாக்கப்பட்ட பின் **getvalue()**-வை வெளிப்படையாக அழைத்து பொருளைத் தொடங்கி வைக்க வேண்டும்.

14.12.2 தானமைவு ஆக்கிகள் :-

அளபுருக்களை ஏற்காத ஆக்கி தானமைவு ஆக்கி என்றழைக்கப்படுகிறது. எடுத்துக்காட்டாக Data என்கிற இனக்குழுவில் Data () என்பது தானமைவு ஆக்கி. இந்த ஆக்கியை பயன்படுத்தி பொருள்களை உருவாக்குவது, பிற தரவு வகைகளைக் கொண்ட மாறிகளை உருவாக்குவது போன்றதாகும்.

எடுத்துக்காட்டு

```
int num;        // சாதாரண மாறி அறிவப்பு

Data d1;        // பொருள் அறிவிப்பு
```

இனக்குழுவில் வெளிப்படையாக ஆக்கிகள் இல்லாத நிலையில் (பயனர் வரையறுக்கும் ஆக்கி) நிரல் பெயர்ப்பி தானாகவே தானமைவு ஆக்கியை உள்ளார்ந்த (implicitly) inline public உறுப்பாக சேர்த்துக் கொள்ளும்.

Chapter 14.indd 212 03/09/18 5:49 PM

பின்வரும் நிரல் 14.18 நிரல் பெயர்ப்பி உருவாக்கும் ஆக்கியை விளக்குகிறது.

```
#include<iostream>
using namespace std;
class Sample
    {
       int i,j;
       public :
       int k;//no user defined constructor in this program
       void getvalue()//member function
        {
          i=j=k=0;
        }
    };
int main()
{
       Sample  s1; //uses the default constructor generated by the compiler
       s1.getvalue();
       return 0;
}
```

குறிப்பு

பயனர் வரையறுக்கும் ஆக்கி இல்லாதபோது நிரல்பெயர்ப்பி தானாகவே தானமைவு ஆக்கியை உருவாக்குகிறது. இது பொருளுக்குத் தேவையான நினைவகத்தை ஒதுக்குகிறது.

Chapter 14.indd 213 03/09/18 5:49 PM

நிரல் 14.19 இனக்குழுவில் உள்ள ஆக்கி மற்றும் பிற உறுப்பு செயற்கூறுகளை விளக்குகிறது.

```
#include<iostream>
using namespace std;
class simple
{
private:
        int a,b;
public:
simple()
        {
a= 0 ;
                b= 0;
                cout<< "\n Constructor of class-simple ";
        }
        void getdata()
        {
                cout<<"\n Enter values for a and b (sample data 6 and  7)... ";
                cin>>a>>b;
        }
        void putdata()
        {
                cout<<"\nThe two integers are .. "<<a<<'\t'<< b<<endl;
                cout<<"\n The sum of the variables "<<a<<" + "<<b<<" = "<<a+b;
        }
};
int main()
{
        simple s;
`       s.getdata();
        s.putdata();
        return 0;
}
```

வெளியீடு :

```
        Constructor of class-simple
        Enter values for a and b (sample data 6 and  7)... 6 7
        The two integers are .. 6       7
        The sum of the variables 6 + 7 = 13
```

14.12.3 அளபுரு ஏற்கும் ஆக்கி (Parameterized)

அளபுருக்களை ஏற்கும் ஆக்கி, அளபுரு ஏற்கும் ஆக்கியாகும். வேறுபட்ட தொடக்க மதிப்புகளைக் கொண்ட பொருள்களை உருவாக்க இந்த வகை ஆக்கி பயன்படுகிறது. செயற்கூறுக்கு அளபுருக்களை அனுப்புவதன் மூலம் இது சாத்தியமாகிறது.

நிரல் 14.20 அளபுரு ஏற்கும் ஆக்கியை பயன்படுத்தி பொருளை உருவாக்குவதை விளக்குகிறது.

```
#include<iostream>
using namespace std;
class simple
{
private:
  int a,b;
public:
simple(int m, int n)
{
a= m ;
b= n;
cout<< "\n Parameterized Constructor of class-simple "<<endl;
}
void putdata()
{
cout<<"\nThe two integers are .. "<<a<<'\t'<< b<<endl;
cout<<"\n The sum of the variables "<<a<<" + "<<b<<" = "<< a+b;
}
};
int main()
{
simple s1(10,20),s2(30,45); //Created two objects with different values created
cout<<"\n\t\tObject 1\n";
s1.putdata();
cout<<"\n\t\tObject 2\n";
s2.putdata();
return 0;
}
```

வெளியீடு :

```
        Parameterized Constructor of class-simple
        Parameterized Constructor of class-simple
        Object 1
                The two integers are .. 10      20
                The sum of the variables 10 + 20 = 30
        Object 2
                The two integers are .. 30      45
                The sum of the variables 30 + 45 = 75
```

குறிப்பு

அளபுருக்களோடு அறிவிக்கப்படும் ஆக்கி, நிரல்பெயர்ப்பி உருவாக்கும் ஆக்கியை மறைக்கிறது. இதன்பிறகு நிரல் பெயர்ப்பி உருவாக்கும் ஆக்கியை இயக்க முடியாது.

நிரல் 14.21 அளபுருவை ஏற்கும் ஆக்கியை வரையறுத்த பின்னர், அளபுருக்கள் இல்லாத பொருளை உருவாக்கினால் ஏற்படும் பிழையை விளக்குகிறது.

```
#include<iostream>
using namespace std;
class simple
{
private:
  int a,b;
public:
simple(int m, int n)
{
a= m ;
b= n;
cout<< "\n Parameterized Constructor of class-simple "<<endl;
}

void putdata()
{
cout<<"\nThe two integers are .. "<<a<<'\t'<< b<<endl;
cout<<"\n The sum of the variables "<<a<<" + "<<b<<" = "<< a+b;
}
};
int main()
{
simple s,s1(10,20)  // [Error] no matching function for call to 'simple::simple()'
s1.putdata();
s2.putdata();
return 0;
}
```

குறிப்பு

அளபுருக்களை ஏற்கும் ஆக்கி, பிற சாதாரண செயற்கூறுகளைப் போன்று தானமைவு செயலுறுப்புகளைக் கொண்டிருக்கும்.

நிரல் 14.22 அளபுருக்களை ஏற்கும் ஆக்கியில் தானமைவு செயலுருபுகளைப் பற்றி விளக்குதல்

```
#include<iostream>
using namespace std;
class simple
{
private:
  int a,b;
public:
simple(int m, int n=100)    //default argument
{
a= m ;
b= n;
cout<< "\n Parameterized Constructor with default argument"<<endl;
}
void putdata()
{
cout<<"\nThe two integers are .. "<<a<<'\t'<< b<<endl;
cout<<"\n The sum of the variables "<<a<<" + "<<b<<" = "<< a+b;
}
};
int main()
{
simple s1(10,20),s2(50);
 cout<<"\n\t\tObject 1 with both values \n";
s1.putdata();
cout<<"\n\t\tObject 2 with one value and one deafult value\n";
s2.putdata();
return 0;
}
```

வெளியீடு :

```
        Parameterized Constructor with default argument
        Parameterized Constructor with default argument
        Object 1 with both values
                The two integers are .. 10    20
                The sum of the variables 10 + 20 = 30
        Object 2 with one value and one deafult value
                The two integers are .. 50    100
                The sum of the variables 50 + 100 = 150
```

14.13 தானமைவு ஆக்கியின் தனிச்சிறப்பு :

தானமைவு ஆக்கிகள், குறிப்பிட்ட தொடக்க மதிப்பு இல்லாத பொருள்களை உருவாக்கப் பயன்படுகிறது. பொருள்களின் அணியை உருவாக்கவும் இது பயன்படுகிறது.

Chapter 14.indd 217 03/09/18 5:49 PM

நிரல் 14.23 தானமைவு ஆக்கியில் முக்கியத்துவத்தை விளக்குகிறது

```
#include<iostream>
using namespace std;
class simple
{
private:
int a, b;
public:
simple()      //default constructor
{
a=0 ;
b= 0;
cout<< "\n default constructor"<<endl;
}
int getdata();
};
int simple :: getdata()
{  int tot;
cout<<"\nEnter two values ";
cin>>a>>b;
tot=a+b;
return tot;
}
int main()
{
int sum=0;
simple s1[3];
cout<<"\n\t\tObject 1 with both values \n";
for (int i=0;i<3;i++)
   sum+=s1[i].getdata();
cout<<"\nsum of all object values is"<<sum;
return 0;
}
```

வெளியீடு :

```
        default constructor
        default constructor
        default constructor
        Object 1 with both values
                Enter two values 10 20
                Enter two values 30 40
                Enter two values 50 50
                sum of all object values is 200
```

14.14 ஆக்கிகளை அழைத்தல்

இரண்டு வழிகளில் அளபுருக்களை ஏற்கும் ஆக்கியைப் பயன்படுத்திப் பொருளை உருவாக்கலாம்.

Chapter 14.indd 218 03/09/18 5:49 PM

1. உள்ளார்ந்த அழைப்பு (Implicit call)

2. வெளிப்படையான அழைப்பு (Explicit call)

14.14.1 உள்ளார்ந்த அழைப்பு:- (Implicit call)

இந்த முறையில், எப்பொழுதெல்லாம் பொருள் உருவாக்கப்படுகிறதோ அப்பொழுது அளபுருக்களை ஏற்கும் ஆக்கி தானாகவே அழைக்கப்படுகிறது. எடுத்துக்காட்டாக simple s1 (10, 20); இதில் s1 என்ற பொருளை உருவாக்கும்போது அளபுருவை ஏற்கும் ஆக்கி தானாகவே அழைக்கப்படுகிறது.

14.14.2 வெளிப்படையான அழைப்பு :- (Explicit call)

இந்த முறையில் ஆக்கியின் பெயரை வெளிப்படையாக கொடுக்கும்போது அளபுருக்களை ஏற்கும் ஆக்கி அழைக்கப்படுகிறது. அதனால் பொருளை உருவாக்கி தொடங்கி வைக்க முடிகிறது.

எடுத்துக்காட்டாக,

```
simple s1=simple(10,20);        //வெளிப்படையான அழைப்பு
```

வெளிப்படையான அழைப்பு முறை மிகவும் பொருத்தமானது. ஏனென்றால் இதில் தற்காலிக பொருளை உருவாக்குவதால் தரவு இழப்புக்கான வாய்ப்பு ஏற்படாது. தற்காலிக பொருள் கோவையில் பயன்படுத்தும் வரை நினைவகத்தில் இருக்கும். அதன்பிறகு அது அழிந்து விடும்.

குறிப்பு

ஆக்கிகளை வெளிப்படையாக அழைக்கும்போது உருவாகும் தற்காலிக சான்றுரு, நினைவகத்தில் அதைப் பயன்படுத்தும் வரை இருக்கும். பிறகு நினைவகத்திலிருந்து விடுவிக்கப்படும்.

நிரல் 14.24 உள்ளார்ந்த மற்றும் வெளிப்படையான அழைப்பு – விளக்குதல்

```
#include<iostream>
using namespace std;
class simple
{
private:
int a, b;
public:
simple(int m,int n)
{
a= m ;
b= n;
cout<< "\n Constructor of class-simple invoked for implicit and explicit call"<<endl;
}
void putdata()
{
cout<<"\nThe two integers are .. "<<a<<'\t'<< b<<endl;
cout<<"\n The sum of the variables "<<a<<" + "<<b<<" = "<<a+b;
}
};
```

```
int main()
{
simple s1(10,20);              //implicit call
simple s2=simple(30,45);       //explicit call
cout<<"\n\t\tObject 1\n";
s1.putdata();
s2.putdata();
return 0;
}
```

ஆக்கியை வெளிப்படையாக அழைக்கும் போது தற்காலிக சான்றுருவை உருவாக்கும்

வெளியீடு :

```
Constructor of class-simple invoked for implicit and explicit call
Constructor of class-simple invoked for implicit and explicit call
Object 1
        The two integers are .. 10      20
        The sum of the variables 10 + 20 = 30
Object 2
        The two integers are .. 30      45
        The sum of the variables 30 + 45 = 75
```

14.15 நகல் ஆக்கிகள் :

ஏற்கெனவே உள்ள இனக்குழுவின் பொருளுக்குக் குறிப்புகளைக் கொண்ட ஆக்கிக்கு நகல் ஆக்கி என்று பெயர். மாறாக, நகல் ஆக்கி என்பது ஆக்கியின் ஒரு வகை, இது ஏற்கெனவே உள்ள இனக்குழுவின் பொருளை நகலெடுக்கப் பயன்படுகிறது. இதன் பொதுவான அமைப்பு இவ்வாறு இருக்கும் **simple (simple &);** இதில் **simple** என்பது இனக்குழுப் பெயர். நிரல்பெயர்ப்பி, அனைத்து இனக்குழுவிற்கும் தானமைவு நகல் ஆக்கியைக் கொடுக்கிறது.

14.15.1 நகல் ஆக்கிகளை அழைத்தல்

நகல் ஆக்கி கீழ்காணும் சூழல்களில் இயக்கப்படும்.

1) ஏதேனும் ஓர் உறுப்புச் செயற்கூறினுக்கு ஒரு பொருளை அளபுருவாக அனுப்பி வைக்கும் போது.

எடுத்துக்காட்டு : void Simple :: putdata (simple x);

2) ஓர் உறுப்புச் செயற்கூறு ஒரு பொருளைத் திருப்பியனுப்பும் போது.

எடுத்துக்காட்டு : Simple getdata ()

3) ஒரு பொருளைக் குறிப்புவகையாக இனக்குழுவின் சான்றுருக்கு அனுப்பி வைக்கப்படும்போது. எடுத்துக்காட்டு : Simple S1, S2(S1) //S2(S1) நகல் ஆக்கியை அழைக்கிறது.

Chapter 14.indd 220 03/09/18 5:49 PM

14.25 நகலாக்கியின் செயல்பாட்டை விளக்கும் நிரல்

```
#include <iostream>
using namespace std;
class Test
{
  private:
int X;
int Y;
  public:
        Test (int , int );   //parameterized constructor declaration
        Test (Test &);   //Declaration of copy constructor to initialize data
members.
void   Display();
};//End of class
Test:: Test(int a, int b)     //Definition of parameterized constructor.
{
   X = a;
   Y = b;
}
Test::Test(Test &T)     //Definition of copy constructor.
{
   X = T.X;
   Y = T.Y;
}
void Test:: Display()//Definition of Display () member function.
{
cout<<endl<< "X: " << X;
cout<<endl<< "Y: " << Y <<endl;
}
int main()
{
        Test T1(10,20) ; //Parameterized Constructor  automatically called
when
        //object is created.
cout<<endl<<"T1 Object: " <<endl;
cout<< "Value after initialization : " ;
   T1.Display();
   Test T2(T1);//Intialize object with other object using copy constructor
cout<<endl<< "T2 Object: " <<endl;
cout<< "Value after initialization : ";
   T2.Display();
   return 0;
}
```


Chapter 14.indd 221 03/09/18 5:49 PM

வெளியீடு :

```
T1 Object:
Value after initialization :
        X: 10
        Y: 20
T2 Object:
Value after initialization :
X: 10
Y: 20
```

குறிப்பு

நகல் ஆக்கியை வரையறுக்கும் போது செயலுறுப்புகளை (பொருள்) குறிப்பு முறையில் அனுப்ப வேண்டுமே தவிர மதிப்பு மூலம் அல்ல.

14.16 ஆக்கியை இயக்கும் வரிசை முறை

ஆக்கிகள் பொருள் அறிவிப்பின் வரிசைப்படி செயல்படுத்தப்படுகிறது. (ஒரே கூற்றில் இருந்தால் இடமிருந்து வலப்புறமாக செயல்படும்)

எடுத்துக்காட்டாக,

Test t1; //ஆக்கியின் செயல்பாடு முதலில் t1 பிறகு t2 என செயல்படுத்தப்படும்.

Test t2;

பின்வரும் எடுத்துக்காட்டைக் கவனி,

Sample s1,s2,s3; //ஆக்கியின் வரிசை s1 பிறகு s2 மற்றும் இறுதியாக s3.

ஆனால் ஒரு இனக்குழு மற்றொரு இனக்குழுவின் பொருளை உறுப்பாக கொண்டிருந்தால் பிறகு அந்த இனக்குழுவின் ஆக்கி முதலில் செயல்படுத்தப்படும்.

நிரல் 14.27 ஆக்கியின் செயல்படு வரிசைமுறையை விளக்குகிறது

```
#include<iostream>
using namespace std;
class outer
  {
int data;
    public:
         outer()
         { cout<<"\nconstructor  of class outer ";
         }
    };
```


Chapter 14.indd 222 03/09/18 5:49 PM

```
class inner
  {
          outer ot;       // object ot of class outer is declared in class inner
    public:
          inner()
          { cout<<"\n constructor  of class inner ";}
  };
int main()
{
    inner in;
     return 0;
}
```

வெளியீடு :

```
          constructor  of class outer
          constructor  of class inner
```

14.17 பொருள் இயங்குநிலையில் தொடங்குதல்

இயங்கு நேரத்தில் தொடக்க மதிப்புகள் கொடுக்கப்பட்டால் அது இயங்கு நிலையில் தொடங்குதல் எனப்படும்.

நிரல் 14.28 இயங்குநிலை தொடங்கலை எடுத்துக்காட்டுகிறது

```
#include<iostream>
using namespace std;
class X
  {
int n;
    float avg;
    public:
          X(int p,float q)
          {  n=p;
          avg=q;
                    }
void disp()
{
          cout<<"\n Roll numbe:- " <<n;
          cout<<"\nAverage :- "<<avg;
}
    };
```

நிரல் 14.28 இயங்குநிலை தொடங்கலை எடுத்துக்காட்டுகிறது

```
int main()
{
int a ; float b;
        cout<<"\nEnter the Roll Number";
        cin>>a;
        cout<<"\nEnter the Average";
        cin>>b;
         X x(a,b);       // dynamic initialization
        x.disp();
         return 0;
}
```

வெளியீடு :

```
        Enter the Roll Number 1201
        Enter the Average 98.6
        Roll numbe:- 1201
        Average :- 98.6
```

14.18 ஆக்கியின் தனிச்சிறப்புப் பண்புகள்

- ஆக்கியின் பெயர் இனக்குழுவின் பெயராகவே இருக்க வேண்டும்.
- ஆக்கி எந்த மதிப்பையும் திருப்பி அனுப்பாது.
- ஆக்கி அளபுருக்களின் பட்டியலைக் கொண்டிருக்கும்.
- ஆக்கி செயற்கூறு, பணி மிகுக்கப்பட முடியும்.
- ஆக்கியைத் தருவிக்க முடியாது. ஆனால் தருவிக்கப்பட்ட இனக்குழு அடிப்படை இனக்குழுவின் ஆக்கியை அழைக்க முடியும்.
- பயனர் வரையறுக்கும் ஆக்கி இல்லாத போது நிரல்பெயர்ப்பி ஓர் ஆக்கியை உருவாக்கிக் கொள்ளும்.
- நிரல்பெயர்ப்பி உருவாக்கும் ஆக்கி Public உறுப்பு செயற்கூறாகும்.
- ஓர் இனக்குழு பொருள் உருவாக்கப்படும்போது ஆக்கி தானாகவே இயக்கப்படும்.
- ஒரு புதிய பொருளை அதன் இனக்குழுவில் உருவாக்க ஆக்கியை வெளிப்படையாகப் பயன்படுத்த முடியும்.

14.19 அழிப்பிகள்

அழிப்பி என்பது, ஒரு பொருளை உருவாக்கும்போது, ஆக்கியால் பொருளுக்கென ஒதுக்கப்படும் நினைவகப் பகுதியை விடுவிக்கும் ஒரு செயற்கூறாகும். இதுவும் இனக்குழுவின் பெயரையேக்

கொண்டிருக்கும். ஆனால் ~ என்னும் குறியைப் பெயரின் முன்னொட்டாக கொண்டிருக்கும். இது எந்த மதிப்பையும் திருப்பி அனுப்புவதில்லை, எந்த தரவினத்தோடும் தொடர்புடையவை அல்ல.

14.19.1 அழிப்பியின் தேவை

அழிப்பியின் நோக்கம் ஒரு பொருள் அதன் வாழ்நாளில் பெற்ற வளங்களை விடுவிப்பதாகும். ஒரு பொருளை உருவாக்கும் போது ஆக்கியால் பொருளுக்கென ஒதுக்கப்பட்ட நினைவகப் பகுதியை அழிக்கும்.

14.20 அறிவிப்பு மற்றும் வரையறுப்பு

அழிப்பி என்ற சிறப்பு செயற்கூறானது ஆக்கியால் உருவாக்கப்பட்ட பொருளின் வாழ்நாள் முடிந்து அழியும்போது அழைக்கப்படுகிறது. பொதுவாக Public பகுதியில் அறிவிக்கப்படுகிறது.

நிரல் 14.29 அழிப்பி செயற்கூறினை விளக்குகிறது

```
#include<iostream>
using namespace std;
class simple
{
private:
int a, b;
public:
simple()
{
a= 0 ;
b= 0;
cout<< "\n Constructor of class-simple ";
}
void getdata()
{
cout<<"\n Enter values for a and b (sample data 6 and  7)... ";
cin>>a>>b;
}
void putdata()
{
cout<<"\nThe two integers are .. "<<a<<'\t'<< b<<endl;
cout<<"\n The sum of the variables "<<a<<" + "<<b<<" = "<<a+b;
}
~simple()
{    cout<<"\n Destructor is executed to destroy the object";}
};
int main()
{
simple s;
s.getdata();
s.putdata();
return 0;
}
```


Chapter 14.indd   225
03/09/18   5:49 PM

வெளியீடு :

Constructor of class-simple

Enter values for a and b (sample data 6 and 7)... 6 7

The two integers are .. 6 7

The sum of the variables 6 + 7 = 13

Destructor is executed to destroy the object

14.21 அழிப்பியின் தனிச் சிறப்புப் பண்புகள்

1) அழிப்பியின் பெயரானது ~ என்ற முன்னொட்டு குறியுடன் கூடிய இனக்குழுவின் பெயரையேக் கொண்டிருக்கும்.

2) அழிப்பி, செயலுருபுகளை ஏற்காது.

3) அழிப்பி எந்த மதிப்பையும் திருப்பி அனுப்பாது.

4) அழிப்பி பணிமிகுக்கப்பட முடியாது. அதாவது ஓர் இனக்குழுவில் ஓர் அழிப்பி மட்டுமே இருக்க முடியும்.

5) பயனர் அழிப்பியை வரையறுக்காத போது நிரல்பெயர்ப்பி ஓர் அழிப்பியை உருவாக்கிக் கொள்ளும்.

6) நிரலில் உருவாக்கப்பட்ட ஓர் இனக்குழு பொருளின் பயன்பாடு முடிவுக்கு வரும்போது அழிப்பி தானாகவே இயக்கப்படும்.

7) இதை தருவிக்க (மரபுவழி) முடியாது.

நினைவில்கொள்க

- தரவையும் அதனோடு தொடர்புடைய செயற்கூறினையும் இணைத்து வைப்பது இனக்குழுவாகும்.
- C++ -ல் உள்ள பயனர் வரையறுக்கும் தரவினமான இனக்குழுவைப் பயன்படுத்திப் பொருளை உருவாக்கலாம்.
- ஓர் இனக்குழுவை அறிவிக்கும் பொழுது தரவு உறுப்புகள், உறுப்பு செயற்கூறுகள், அணுகியல்பு வரையறுப்பிகள் மற்றும் இனக்குழுவின் பெயர் கொடுக்கப்படுகிறது.
- ஓர் இனக்குழுவின் உறுப்பு செயற்கூறினை அந்த இனக்குழுவின் உள்ளே அல்லது வெளியே வரையறுக்கலாம்
- இனக்குழுவின் public உறுப்புகளை அந்த இனக்குழு பொருளைப் பயன்படுத்தி இனக்குழுவின் வெளியிலிருந்து நேரடியாக அணுகலாம்.
- ஓர் இனக்குழுவின் OOP – சிறப்பம்சமான உறைப்பொதியாக்கத்தைத் தரவு மற்றும் தொடர்புடைய செயற்கூறுகளை ஒன்றாக சேர்த்து வைப்பதன் மூலம் ஆதரிக்கிறது.
- Private மற்றும் protected மூலம் வெளி உலகிற்கு தகவலை மறைப்பதன் மூலம் தரவு மறைத்தலை இனக்குழு ஆதரிக்கிறது.

நினைவில்கொள்க

- ஒரு இனக்குழுவின் உறுப்பு செயற்கூறினை மற்றொரு உறுப்பு செயற்கூறு மூலம் அழைத்தல் பின்னலான உறுப்பு செயற்கூறு என்றழைக்கப்படுகிறது.
- வரையெல்லை உணர்த்தும் செயற்குறி ::, ஒரு இனக்குழு பெயரில் பயன்படுத்தும் class-name :: function name பயன்படுத்தும் போது அந்த இனக்குழுவின் உறுப்புகளைக் குறிக்கிறது. :: variable-name, முழுதளாவிய மாறியைக் குறிக்கிறது.
- ஓர் இனக்குழுவின் சான்றுரு பயன்பாட்டுக்கு வரும்போது ஆக்கி எனப்படும் சிறப்பு செயற்கூறு இயக்கப்படுகிறது.
- ஆக்கி செயற்கூறு இனக்குழு பொருளுக்கு நினைவகத்தில் இடம் ஒதுக்குகிறது மற்றும் தொடங்கி வைக்கிறது.
- அழிப்பி என்ற சிறப்பு செயற்கூறானது ஆக்கியால் உருவாக்கப்பட்ட பொருளின் வாழ்நாள் முடிந்து அழியும்போது அழைக்கப்படுகிறது.
- ஆக்கிகள் மற்றும் அழிப்பிகள் பெயர் இனக்குழுவின் பெயராக இருக்க வேண்டும்.
- அளபுருக்கள் இல்லாத ஆக்கி தானமைவு ஆக்கி என்றழைக்கப்படுகிறது.
- தானமைவு செயலுருபுகளைக் கொண்ட ஆக்கி தானமைவு ஆக்கிக்கு சமமானதாகும்.
- ஆக்கி மற்றும் அழிப்பி எதையும் திருப்பி அனுப்பாது. எந்த தரவினத்தோடும் தொடர்புடையது அல்ல.
- பொருளை இயங்குநிலையில் தொடங்க முடியும்.

பயிற்சி

1 பின்வரும் குறிப்புகளைப் பயன்படுத்தி Employee இனக்குழுவை வரையறுக்கவும்

Employee-இனக்குழுவின் private உறுப்புகள்

empno- integer

ename – 20 characters

basic – float

netpay, hra, da, - float

calculate () – basic+hra+da - வைக் கண்டுப்பிடித்து float மதிப்பைத் திருப்பி அனுப்பும் செயற்கூறு

employee - இனக்குழுவின் public செயற்கூறுகள்

havedata() – (empno, ename, basic, hra, da) உள்ளீடாகப்பெற்று

Chapter 14.indd 227 03/09/18 5:49 PM

netpay கணக்கிட calculate() செயற்கூறை அழைக்கும் ஒரு செயற்கூறு

dispdata() – அனைத்து தரவு உறுப்புகளையும் திரையில் வெளியடுவதற்கான செயற்கூறு

2. பின்வரும் குறிப்புகளைப் பயன்படுத்தி MATH- இனக்குழுவை வரையறுக்கவும்

private உறுப்புகள்

num1, num2, result – float

init() செயற்கூறு num1, num2, result - க்கு தொடக்க மதிப்பிருத்தப் பயன்படுகிறது

protected உறுப்புகள்

num1 & num2 -வைக் கூட்டி, வரும் விடையை sum-ல் இருத்த பயன்படும் add() செயற்கூறு

num1 & num2 -வைக் கழித்து, வரும் விடையை result-ல் இருத்த பயன்படும் diff() செயற்கூறு பயன்படும்

public உறுப்புகள்

num1 & num2 மதிப்புகளை உள்ளீடாகப் பெறுவதற்கு getdata() செயற்கூறு பயன்படும்

menu -வை திரையிட menu() செயற்கூறு பயன்படும்

1. Add…

2. Subtract…

invoke add() தேர்வு 1 என இருந்தால் add() செயற்கூறினை அழைக்கும் மற்றும் 2 என இருந்தால் diff() செயற்கூறினை அழைக்கும் பிறகு விடையைத் திரையிடும்.

3. பின்வரும் குறிப்புகளைப் பயன்படுத்தி Item இனக்குழுவை வரையறுக்கவும்

private உறுப்புகள்

code, quantity -Integer

price -Float

அனைத்து தரவு உறுப்புகளின் மதிப்பை உள்ளீடாகப் பெற்று, திருப்பி அனுப்பும் மதிப்பு இல்லாத getdata() செயற்கூறு

public உறுப்புகள்

tax – float

code, quantity, price and tax-திரையில் திரையிட பயன்படும் செயற்கூறு dispdata(). quantity 100 க்கும் அதிகம் என்றால் tax = 2500, இல்லையெனில் tax =1000 எனக் கணக்கிடவும்.

4. FRAME இனக்குழுவின் வரையறையை கீழேக்கொடுக்கப்பட்டுள்ள குறிப்புகளுடன் எழுதவும்.

Private உறுப்புகள்

10*Height*Width மதிப்பைக் கணக்கிட்டு, amount-ல் இருத்தி வைக்கப் பயன்படுத்தப்படும் செயற்கூறு SetAmount( )

Public உறுப்புகள்

FrameID, Height, Width- மதிப்புகளை உள்ளீடு செய்ய GetDetail() செயற்கூறைப் பயன்படுத்தவும்.

இச்செயற்கூறு Amountஐ கணக்கிட SetAmount( ) செயற்கூறை அழைக்கும்.

அனைத்து தரவு உறுப்புகளையும் திரையிடுவதற்கு ShowDetail() செயற்கூறு பயனபடும்.

5. RESORT இனக்குழுவைப் பின்வரும் குறிப்புகளைப் பயன்படுத்தி வரையறுக்கவும்:

Private உறுப்புகள்

Rno // அறை எண்ணை இருத்தி வைக்கும் தரவு உறுப்பு

Name //பயனரின் பெயரை இருத்தி வைக்கும் தரவு உறுப்பு

Charges // ஒரு நாளின் கட்டணத்தை இருத்தி வைக்கும் தரவு உறுப்பு

Days // நாளின் எண்ணிக்கையை இருத்தி வைக்கும் தரவு உறுப்பு

Compute ( ) // Days * Charges கணக்கீடு செய்யும் செயற்கூறு மொத்த தொகை 5000க்கும் மேல் இருந்தால், மொத்த தொகையைக் கணக்கிட 1.02 * Days *Charges

Public உறுப்புகள்

Getinfo() // Rno, Name, Charges, Days - உள்ளீடாகப் பெற உதவும் செயற்கூறு.

Displayinfo() Rno, RName, Charges, Days Amount திரையில் திரையிட பயன்படும் செயற்கூறு (Compute ( ) செயற்கூற்றை அழைப்பதன் மூலம் Amount கணக்கிடப்படும்.)

6. structpno

```
{
    int pin;
    floatbalane;
    }
```

BankAccount இனக்குழுவைப் பின்வரும் குறிப்புகளுடன் வரையறு

Protected உறுப்புகள்

பத்து மதிப்புகளைக் கொண்ட அணி init pin உள்ளீடாகப்பெற்று, தொடக்க மதிப்பிருத்தி, balance amount- மதிப்பு 0 என இருத்தவும்.

public உறுப்புகள்

deposit(pin, amount):

amount மற்றும் pin உள்ளீடாகப் பெற்று account balance -ன் மதிப்பை மிகுக்கவும். pin மதிப்பை பெற்று பரிசோதிக்கும். சரி எனில் balanceஐ மிகுக்கவும், திரையில் balanceஐ திரையிடவும் இல்லையெனில் பொருத்தமான செய்தியைத் திரையிடும்.

withdraw(self, pin, amount):

amount மற்றும் pin ஐ உள்ளீடாகப் பெற்று, account balance -ஐ குறைக்கவும். pin ஐ சரிபார்க்கவும் மீதித்தொகை 1000க்கும் அதிகமாகவும் மற்றும் amount, balance -ஐ விட குறைவாகவும் இருக்க வேண்டும். சரி எனில் amount ஐ எடுக்கவும், மேலும் balance ஐ திரையிடவும் அல்லது பொருத்தமான செய்தியை வெளியிடவும்.

7. Hotel இனக்குழுவைப் பின்வரும் குறிப்புகளுடன் C++ -ல் வரையறுக்கவும்

Private உறுப்புகள்

Rno // தரவு உறுப்பைச் சேமிக்க

Name // தரவு உறுப்பைச் சேமிக்க

Charges // தரவு உறுப்பைச் சேமிக்க

Days // தரவு உறுப்பைச் சேமிக்க

Calculate() // Amount ஐ கணக்கிட்டு திருப்பி அனுப்பும் செயற்கூறு

//Days*Charges கணக்கீடு செய்து Amount -ல் மதிப்பைத் திருப்பி அனுப்புதல் Days*Charges 12000 க்கும் அதிகம் எனில் 1.2*Days*Charges

Public உறுப்புகள்

Hotel() // இனக்குழு உறுப்புகளைத் தொடக்க மதிப்பிருத்தல்

Getinfo() // Rno, Name, Charges, Days உள்ளீடாகப் பெற பயன்படும் செயற்கூறு

Showinfo() // Rno, RName, Charges, Days திரையில் திரையிட பயன்படும் செயற்கூறு

//Amount (Amount -யைத் திரையிட Calculate( ) செயற்கூறை அழைத்தல்)

8. Exam இனக்குழுவைப் பின்வரும் குறிப்புகளுடன் C++ வரையறு

Private உறுப்புகள்

Rollno -Integer

Cname -25 characters

Mark -integer

public :

Exam(int,char[],int) // பொருளுக்குத் தொடக்க மதிப்பிருத்தல்

~Exam( ) // "Result will be intimated shortly" செய்தியைத் திரையிடும்

void Display( )// mark 60-க்கு மேல் என்றால் அனைத்து தகவல்களும் திரையிடப்படும்

இல்லையெனில் “Result Withheld” செய்தியைத் திரையிடுவதற்கு பயன்படும் செயற்கூறு

9. Student இனக்குழுவைப் பின்வரும் குறிப்புகளைப் பயன்படுத்தி வரையறு:

Private உறுப்புகள்

தரவு உறுப்பு Rno- long வகை

தரவு உறுப்பு Cname-string வகை

தரவு உறுப்பு Agg_marks -float வகை

தரவு உறுப்பு Grade -char வகை

உறுப்பு செயற்கூறு setGrade ()- மாணவன் பெற்ற மதிப்பெண் அடிப்படையில் grade கண்டுபிடிக்கவும்.

மதிப்பெண் சராசரியும்	Grade -யும் பின்வருமாறு காண்க.
>=90	A
Less than 90 and >=75	B
Less than 75 and >=50	C
Less than 50	D

Public உறுப்புகள்

தரவு உறுப்புகளுக்கு ஆக்கி மூலம் மதிப்பிருத்தல், நகல் ஆக்கி மூலம் ஒரு பொருளின் மதிப்பை மற்றொரு பொருளுக்கு இருத்தல் Rno.Cname, Agg_marks-மதிப்புகளை உள்ளீடு செய்வதற்கு பயன்படும் செயற்கூறு மற்றும் functionsetGrade ()-grade-யைக் கண்டுப்பிடிப்பதற்கு அழைத்தல் அனைத்து தரவு உறுப்புகளின் மதிப்பை காண பயன்படுத்தும் செயற்கூறு, END செய்தியைத் திரையிடும் ஓர் அழிப்பி.

Rno=0,Cname=”N.A”,Agg_marks=0.0

மதிப்பீடு

N24N4N

பகுதி அ

I சரியான விடை யை தேர்வு செய்யவும் :

1. ஓர் இனக்குழுவுக்குள் அறிவிக்கப்படும் மாறிகளை தரவு உறுப்புகள் என குறிப்பிடுகின்றோம். செயல்கூறுகளை எவ்வாறு குறிப்பிடுகிறோம்.

(அ) தரவு செயற்கூறுகள் (ஆ) inline செயற்கூறுகள்

(இ) உறுப்பு செயற்கூறுகள் (ஈ) பண்புக் கூறுகள்

2. பின்வரும் உறுப்புச் செயற்கூறினைப் பற்றிய கூற்றுகளில் எது சரி அல்லது தவறு ?

i) புள்ளி செயற்குறி மூலம் ஒரு உறுப்புச் செயற்கூறு, இன்னொரு உறுப்புச் செயற்கூறினை நேரடியாக

Chapter 14.indd 231 03/09/18 5:49 PM

அழைக்கலாம்.

ii) இனக்குழுவின் private தரவுகளை உறுப்புச் செயற்கூறு அணுக முடியும்.

(அ) i - சரி, ii - சரி (ஆ) i – தவறு, ii - சரி

(இ) i -தவறு, ii - சரி (ஈ)i – தவறு, ii – தவறு

3. ஒரு உறுப்பு செயற்கூறு, இன்னொரு உறுப்பு செயற்கூறைப் புள்ளி செயற்குறியைப் பயன்படுத்தாமல் நேரடியாக அணுகலாம் என்பதை எவ்வாறு குறிப்பிடலாம்.

(அ) துணை செயற்கூறு (ஆ) துணை உறுப்பு

(இ) பின்னலான உறுப்பு செயற்கூறு (ஈ) துணை உறுப்பு செயற்கூறு

4. இனக்குழுவுக்குள் வரையறுக்கப்படும் செயற்கூறுகள் எந்த செயற்கூறுகளைப் போல் இயங்குகின்றன?

(அ) inline செயற்கூறுகள் (ஆ) inline அல்லாத செயற்கூறுகள்

(இ) Outline செயற்கூறுகள் (ஈ) தரவு செயற்கூறு

5. பின்வரும் எந்த அணுகியல்பு வரையறுப்பி தவறுதலான மாற்றங்களிலிருந்து தரவைப் பாதுகாக்கிறது?

(அ) Private (ஆ) Protected (இ) Public (ஈ) முழுதளாவிய

6. கீழ்கண்ட நிரலில் எத்தனை பொருள்கள் உருவாக்கப்பட்டுள்ளன ?

```
class x
{
int y;
public:
x (int z) {y=z;}
} x1 [4];
int main ()
{ x x2(10);
return 0;}
```

(அ) 10 (ஆ) 14 (இ) 5 (ஈ) 2

7. ஆக்கி செயற்கூறு பற்றிய பின்வரும் கூற்றுகள் சரியா, தவறா எனக் கூறு.

i) ஆக்கிகள் private பகுதியில் அறிவிக்கப்பட வேண்டும்

ii) பொருள்கள் உருவாக்கப்படும் போது, ஆக்கி தானாகவே இயக்கப்படும்.

(அ) சரி, சரி (ஆ) சரி, தவறு (இ) தவறு, சரி (ஈ) தவறு தவறு

8. பின்வரும் முன்வடிவுக்கு கீழ்கண்டவற்றுள் எந்த ஆக்கி இயக்கப்படும் ?

add display (add &); //add என்பது இனக்குழுவின் பெயர்

(அ) தானமைவு ஆக்கி (ஆ) அளபுருக்களுடன் கூடிய ஆக்கி

(இ) நகல் ஆக்கி (ஈ) அளபுருக்கள் இல்லாத ஆக்கி

9. ஒரு நிரலில், இனக்குழு அளபுருக்களுடன் கூடிய ஆக்கியை பெற்று, ஆனால் தானமைவு ஆக்கி இல்லாத போது அளபுருக்கள் இல்லாத ஆக்கியைக் கொண்ட பொருளை உருவாக்கினால் என்னவாகும் ?

(அ) நிரல் பெயர்ப்பி-நேரப்பிழை (ஆ) களப்பிழை

(இ) நிகழ் நேரப் பிழை (ஈ) நிகழ்நேர விதிவிலக்கு

10. பின்வருவனவற்றுள் எது தற்காலிக சான்றுருவை உருவாக்கும் ?
(அ) ஆக்கியின் உள்ளார்ந்த அழைப்பு (ஆ) ஆக்கியை வெளிப்படையாக அழைத்தல்
(இ) அழிப்பியின் உள்ளார்ந்த அழைப்பு (ஈ) அழிப்பியை வெளிப்படையாக அழைத்தல்

பகுதி– ஆ

II குறுவினாக்கள் (2 மதிப்பெண்கள்)

1. உறுப்புகள் என்றால் என்ன?

2. பயனர் வரையறுத்த தரவினம் வகையான கட்டுரு, இனக்குழு – வேறுபடுத்திக் காட்டுக.

3. பொருள் நோக்கு நிரலாக்கு குறிமுறை **(OOP)** அடிப்படையில் இனக்குழு மற்றும் பொருள் பற்றி வேறுபடுத்திக் காட்டுக.

4. நிரல்பெயர்ப்பி தாமாகவே ஆக்கியை உருவாக்கிக் கொள்ள முடிந்தாலும், ஆக்கி வரையறுப்பு ஏன் சிறந்த வழக்கம் என்று கருதப்படுகிறது ?

5. அழிப்பியின் முக்கியத்துவத்தைப் பற்றி எழுதுக.

பகுதி– இ

III சிறுவினாக்கள் (3 மதிப்பெண்கள்)

1. பின்வரும் நிரலில் கட்டளை அமைப்புப் பிழை ஏதேனும் இருப்பின், அவற்றை நீக்கி, பிழையைக் கோடிட்டு காட்டி, நிரலை மாற்றி எழுதவும்.

```
#include<iostream>
#include<stdio.h>
  class mystud
  { int studid =1001;
  char name[20];
public
  mystud( )
  { }
void register ( ) {cin>>stdid;gets(name);
}
void display ( )
  { cout<<studid<<”: “<<name<<endl;}
}
int main( )
  { mystud MS;
  register.MS( );
  MS.display( );
```

```
}
```

2. நிரலின் இயங்கு நேரத்தில் ஒரு பொருளை எவ்வாறு தொடங்கி வைப்பது என்பதை ஒரு எடுத்துக்காட்டுடன் எழுது.
3. Public அணுகுமுறையில் ஆக்கிகள், அழிப்பிகள் அறிவிப்பினால் விளையும் நன்மைகள் யாவை?
4. கீழே கொடுக்கப்பட்டுள்ள C++ நிரலைக் கொண்டு (i) & (ii) வினாக்களுக்கான விடைகளைத் தருக.

```
class TestMeOut
{
public:
  ~TestMeOut() //Function 1
  {cout<<"Leaving the examination hall"<<endl;}
  TestMeOut() //Function 2
  {cout<<"Appearing for examination"<<endl;}
void MyWork() //Function 3
  {cout<<"Attempting Questions//<<endl;}
};
```

(i) பொருள்நோக்கு நிரலாக்க முறையின்படி, செயற்கூறு-1 என்பது எதைக் குறிக்கிறது, எப்பொழுது அது அழைக்க / இயக்கப்படுகிறது ?

(ii) பொருள்நோக்கு நிரலாக்க முறையின்படி, செயற்கூறு-2 என்பது எதைக் குறிக்கிறது, எப்பொழுது அது இயக்க / அழைக்கப்படுகிறது ?

5. பின்வரும் சி++ நிரல் குறிமுறைக்கு வெளியீட்டு எழுது.

```
#include<iostream>
using namespace std;
class Calci
{
char Grade;
int Bonus;
public:
  Calci() {Grade='E'; Bonus=0;}      //ascii value of A=65
void Down(int G)
{
  Grade-=G;
}
```

```
void Up(int G)
{
  Grade+=G;
  Bonus++;
}
void Show()
{
  cout<<Grade<<"#"<<Bonus<<endl;
  }
};
int main()
{
Calci c;
c.Down(3);
c.Show();
c.Up(7);
c.Show();
c.Down(2);
c.Show();
return 0;
}
```

பகுதி – ஈ

IV பெருவினாக்கள் (5 மதிப்பெண்கள்)

1. பின்னலான இனக்குழுவை எடுத்துக்காட்டுடன் விளக்கு
2. ஆக்கி, அழிப்பி – வேறுபாடு தருக
3. கீழ்காணும் வரையறுப்புகளுடன் Resort என்னும் ஓர் இனக்குழுவை வரையறுக்கவும்

private உறுப்புகள்

Rno // அறை எண்ணை இருத்தி வைக்கும் தரவு உறுப்பு

Name //பயனரின் பெயரை இருத்தி வைக்கும் தரவு உறுப்பு

Charges // ஒரு நாளுக்குரிய கட்டணத்தை இருத்தி வைக்கும் தரவு உறுப்பு

Days // நாட்களின் எண்ணிக்கையை இருத்தி வைக்கும் தரவு உறுப்பு

Compute ( ) // Days * Charges கொண்டு மொத்த தொகையை கணக்கிடும் செயற்கூறு

// மொத்த தொகை 11000 ரூபாய்க்கு மேல் இருந்தால், மொத்த தொகையைக் கணக்கிட 1.02 * Days *Charges

Public member:

getinfo( ) // பெயர், அறைஎண், கட்டணம், நாட்கள் போன்ற தகவல்களை உள்ளீடாகப் பெறும் செயற்கூறு

dispinfo ( ) // உள்ளிடப்பட்ட தரவுகள் மற்றும் Compute செயற்கூறினைப் பயன்படுத்தி கணக்கிட மொத்த தொகையை வெளியிடும் செயற்கூறு

4. கீழ்காணும் நிரலுக்கு வெளியீடு எழுது.

```
#include<iostream>
#include<stdio.h>
using namespace std;
class sub
{
int day, subno;
public :
  sub(int,int);    // prototype
void  printsub()
  { cout<<" subject number : "<<subno;
  cout<<" Days : " <<day;
  }
};
  sub::sub(int d=150,int sn=12)
  { cout<<endl<<"Constructing the object "<<endl;
  day=d;
  sub no=sn;
}
class stud
{
  int rno;
  float marks;
public:
  stud( )
  { cout<<"Constructing the object of students "<<endl;
  rno=0;
  marks=0.0;
}
```


Chapter 14.indd 236 03/09/18 5:49 PM

```
void getval()
{
  cout<<"Enter the roll number and the marks secured ";
  cin>>rno>>marks;
}
void printdet()
  { cout<<"Roll no : "<<rno<<"Marks : "<<marks<<endl;
  }
};
class addmission {
  sub obj;
  stud objone;
  float fees;
public :
  add mission ( )
  { cout<< "Constructing the object of admission "<<endl;
  fees=0.0;
}
  void printdet( )
  { objone.printdet();
obj.printsub( );
cout<<"fees : "<<fees<<endl ;
}
};
  int main()
  {system("cls");
  addmission adm;
  cout<<endl<< "Back in main ( )";
  return 0; }
```

5. Write the output of the following

```
#include<iostream>
#include<stdio.h>
  using namespace std;
class P
{ public:
```

```
  P ( )
  { cout<< "\nConstructor of class P "; }
  ~ P ( )
  { cout<< "\nDestructor of class P "; }
  };
  class Q
{ public:
  Q( )
  { cout<<"\nConstructor of class Q "; }
  ~ Q( )
  { cout<< "\nDestructor of class Q "; }
  };
  class R
  {  P obj1, obj2;
  Q obj3;
public:
  R ( )
  { cout<< "\nConstructor of class R ";}
  ~ R ( )
  { cout<< "\nDestructor of class R  ";}
  };
  int main ( )
  { Ro R;
  Q oq;
  P op;
  return 0;
}
```


Chapter 14.indd 238	03/09/18 5:49 PM

N28ASK

பல்லுருவாக்கம்

கற்றலின் நோக்கங்கள் :

இந்த பாடப்பகுதியை கற்றபின் மாணவர் அறிந்து கொள்வது

- பணிமிகுப்பு பற்றி புரிந்து கொள்ளுதல்.
- செயற்கூறு பணிமிகுப்பு, ஆக்கி பணிமிகுப்பு, செயற்குறி பணிமிகுப்பு கொண்ட C++ நிரல்களை உருவாக்குதல்.
- பல்லுருவாக்க கருத்துகளைக் கொண்டு நிரல்களை இயக்குதல் மற்றும் பிழைதிருத்தம் செய்தல்.

15.1 அறிமுகம்

பல்லுருவாக்கம் (polymorphism) என்ற சொல் பல வடிவங்கள் (poly – many, morph – shapes) என்னும் பொருளைத் தருகிறது. பல்லுருவாக்கம் என்பது ஒரு பொருள் அல்லது செயற்கூறினை பல்வேறு வடிவங்களில் காண்பிக்க உதவுகிறது.

C++ மொழியில், பல்லுருவாக்கம் செயற்கூறு பணிமிகுப்பு மற்றும் செயற்குறி பணிமிகுப்பு ஆகியவற்றின் மூலம் நிறைவேற்றப்படுகிறது. பணி மிகுப்பு என்பது ஒரே பெயர், ஒன்றுக்கு மேற்பட்ட வெவ்வேறு பொருளைக் அதைப்போல் செயற்கூறு பணிமிகுப்பு என்பது ஒரே செயற்கூறு அல்லது ஒன்றுக்கு மேற்பட்ட பொருளைக் கொண்டதாகும்.

15.2 செயற்கூறு பணிமிகுப்பு (Function overloading)

செய்தி அல்லது தரவினை ஒன்றுக்கு மேற்பட்ட வடிவங்களில் செயலாக்கவல்ல செயற்கூறின் திறனையே செயற்கூறு பணிமிகுப்பு என்கிறோம். வேறு வகையில் கூறினால், பணி மிகுப்பு என்பது இரண்டு அல்லது அதற்கு மேற்பட்ட செயற்கூறுகள் ஒரே பெயரைப் பகிர்ந்து கொண்டு வேறுபட்ட அளபுருக்களை கொண்டிருக்கும். செயற்கூறுகள் ஒரே பெயரை பகிர்ந்து கொள்ளுதலை பணிமிகுப்பு என்றும் இந்த செயல்பாட்டை செயற்கூறு பணிமிகுப்பு என்கிறோம். செயற்கூறின் அளபுருக்களின் எண்ணிக்கை மற்றும் தரவினங்களை செயற்கூறு முன்வடிவு என்கிறோம். பணி மிகுக்கப்பட்ட செயற்கூறினை அழைக்கும் போது, நிரல் பெயர்ப்பி மிகச் சரியான வரையறுப்பை, அழைக்கப்பட்ட செயற்கூறின் அளபுருக்களின் வகையோடு வரையறுக்கப்பட்ட செயற்கூறின் செயலுருபுகளின் வகையோடு ஒப்பிட்டு தீர்மானிக்கும். மிகச் சிறந்த செயற்கூறு அல்லது செயற்குறி பணிமிகுப்பு தேர்ந்தெடுப்பு முறையை பணிமிகுப்பு தீர்மானம் என்கிறோம்.

15.2.1 செயற்கூறு பணிமிகுப்பின் தேவைகள் (Need For Function overloading)

சில சமயங்களில், ஒரே செயலைக் குறிக்கும் வெவ்வேறு பொருள் கொண்ட பெயரைக் கண்டறிதல் கடினமான செயல் ஆகும்.

வட்டம், முக்கோணம் மற்றும் செவ்வகம் ஆகியவற்றின் பரப்பளவைக் கண்டறியும் செயற்கூறு வரையறைகளைக் கீழே காணலாம்.

float area_circle(float radius) // ஒரு வட்டத்தின் பரப்பைக் கணிக்க

float area_triangle(float half,floatbase,float height) // முக்கோணத்தின் பரப்பைக் கணிக்க

float area_rectangle(float length , float breadth)
// செவ்வகத்தின் பரப்பைக் கணிக்க

இதனைக் கீழ்கண்ட விதத்தில் ஒற்றை செயற்கூறு தலைப்புக் கொண்டு மாற்றி எழுதலாம்.

float area (float radius);

float area (float half, float base, float height);

float area (float length , float breadth);

நிரல்15.1 செயற்கூறு பணிமிகுப்பினை விளக்கும் எளிய C++ நிரலை கீழே காணலாம்.

```
#include <iostream>
using namespace std;
void print(int i)
 {cout<< " It is integer " << i <<endl;}
void print(double  f)
{  cout<< " It is float " << f <<endl;}
void print(string c)
{  cout<< " It is string " << c <<endl;}
int main() {
          print(10);
          print(10.10);
          print("Ten");
          return 0;
}
```

வெளியீடு

```
          It is integer 10
          It is float 10.1
          It is string Ten
```

குறிப்பு:-

செயற்கூறு பணிமிகுப்பு, பல்லுருவாக்கத்தை மட்டுமே நடைமுறைப்படுத்தாமல் ஓர் நிரலில் ஒப்பீடுகளின் எண்ணிக்கையைக் குறைத்து, நிரல் வேகமாக செயல்பட உதவுகிறது. நிரலர், அதிக செயற்கூற்றின் பெயர்களை நினைவில் வைத்துக் கொள்வதை தவிர்க்க வழி செய்கிறது.

15.2.2 செயற்கூறு பணிமிகுப்பிற்கான விதிமுறைகள் (Rules for function overloading)

1. பணிமிகுந்த செயற்கூறுகள் முறையான அளபுருக்களின் எண்ணிக்கையிலோ, அல்லது அவற்றின் தரவு இனங்களிலோ வேறுபட்டிருக்க வேண்டும்.
2. பணிமிகுத்த செயற்கூறுகள் திருப்பியனுப்பும் தரவினம் ஒன்றாக இருக்க வேண்டும் என்ற தேவையில்லை.
3. பணிமிகுத்த செயற்கூறுகளின் முன்னியல்பு செயலுருபுகளை அளபுருக்களின் பட்டியலில் ஒரு பகுதியாக C++ நிரல் பெயர்ப்பி கருதிக் கொள்ளாது.

நிரல் 15.2 செயற்கூறு பணிமிகுப்பினை விளக்கும் C++ நிரலை கீழே காணலாம்

```
#include <iostream>
using namespace std;
long add(long, long);
long add(long,long,long);
float add(float, float);
int main()
{
 long a, b, c,d;
 float e, f, g;
 cout << "Enter three integers\n";
cin >> a >> b>>c;
d=add(a,b,c);   //number of arguments different  but  same data type
cout << "Sum of 3 integers: " << d << endl;
```


Chapter 15.indd 240 03/09/18 5:35 PM

```
cout << "Sum of 3 integers: " << d << endl;
cout << "Enter two integers\n";
cin >> a >> b;
c = add(a, b);   //two arguments data type same with above function call and
different with below function call
cout << "Sum of 2 integers: " << c << endl;
cout << "Enter two floating point numbers\n";
cin >> e >> f;
 g = add(e, f);  //two arguments  similar to the above function call but data
type different
cout << "Sum of floats: " << g << endl;
}
long add(long c, long g)
{
 long sum;
 sum = c + g;
 return sum;
}
float add(float c, float g)
{
 float sum;
 sum = c + g;
 return sum;
}
long add(long c, long g,long h)
{
 long sum;
 sum = c + g+h;
 return sum;
}
```

வெளியீடு

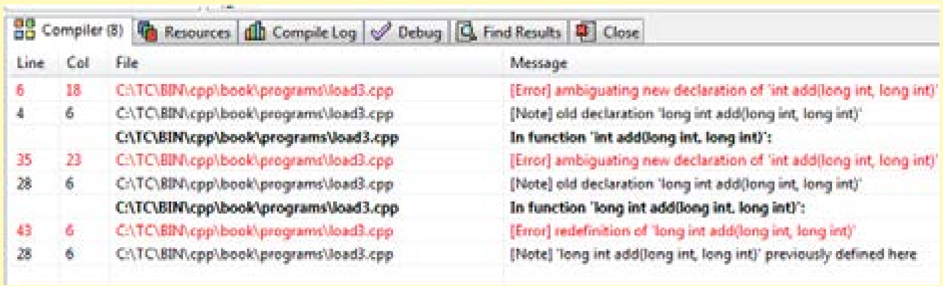
```
Enter three integers
3 4 5
Sum of 3 integers: 12
Enter two integers
4 6
Sum of 2 integers: 10
Enter two floating point numbers
2.1 3.1
Sum of floats: 5.2
```

நிரல் 15.3 முன்னிலை செயலுருபும், தரவினமும் பணிமிகுப்பில் கருத்தில் கொள்ளப்படாது என்பதை விளக்கும் எடுத்துக்காட்டு நிரல்

```
#include <iostream>
using namespace std;
long add(long, long);
long add(long long g=0);
int add(long,long);     //  [Error] ambiguating new declaration of 'int add(long int, long int)
int main()
{
long a, b, c;
int d;
cout<< "Enter two integers\n";
cin>> a >> b;
 d=add(a,b);    //arguments and  datatype are same but return is different
cout<< "Sum of 2 integers: " << d <<endl;
cout<< "Enter two long integers\n";
cin>> a >> b;
 c = add(a, b);
cout<< "Sum of 2 long integers: " << c <<endl;
cout<< "Enter a long integers\n";
cin>> a ;
 c = add(a);     //  [Error] ambiguating new declaration of 'int add(long int, long int)
cout<< "Sum of 2 long integers: " << c <<endl;
}
long add(long c, long g)          //  'long int add(long int, long int)' previously defined here
{
 long sum;
 sum = c + g;
 return sum;
}
int add(long c, long g) //  [Error] ambiguating new declaration of 'int add(long int, long int)
{
int sum;
 sum = c + g;
 return sum;
}
```

```
}
long add(long c, long g=20)     //      [Error] redefinition of 'long int add(long int, long int)'
{
 long sum;
 sum = c + g;
 return sum;
}
```

ERROR Message



Compiler (8) | Resources | Compile Log | Debug | Find Results | Close

Line	Col	File	Message
6	18	C:\TC\BIN\cpp\book\programs\load3.cpp	[Error] ambiguating new declaration of 'int add(long int, long int)'
4	6	C:\TC\BIN\cpp\book\programs\load3.cpp	[Note] old declaration 'long int add(long int, long int)'
		C:\TC\BIN\cpp\book\programs\load3.cpp	In function 'int add(long int, long int)':
35	23	C:\TC\BIN\cpp\book\programs\load3.cpp	[Error] ambiguating new declaration of 'int add(long int, long int)'
28	6	C:\TC\BIN\cpp\book\programs\load3.cpp	[Note] old declaration 'long int add(long int, long int)'
		C:\TC\BIN\cpp\book\programs\load3.cpp	In function 'long int add(long int, long int)':
43	6	C:\TC\BIN\cpp\book\programs\load3.cpp	[Error] redefinition of 'long int add(long int, long int)'
28	6	C:\TC\BIN\cpp\book\programs\load3.cpp	[Note] 'long int add(long int, long int)' previously defined here

15.3 ஆக்கி பணிமிகுப்புConstructor overloading

இனக்குழுவின் சிறப்பு செயற்கூறுகளான ஆக்கிகளையும், செயற்கூறு பணிமிகுப்பு செய்ய முடியும். ஓர் இனக்குழுவில் வெவ்வேறு வரையறுப்புகளைக் கொண்ட ஒன்றுக்கு மேற்பட்ட ஆக்கிகள் இடம் பெறலாம். ஓர் இனக்குழுவிற்கு, பல்வேறு வகையான பொருள்களை உருவாக்க ஆக்கி பணிமிகுப்பு வழி வகை செய்கிறது.

நிரல் 15.4 ஆக்கி பணிமிகுப்பை விளக்கும் நிரல்

```
#include<iostream>
using namespace std;
class add
{
int num1, num2, sum;
public:
add()
{
cout<<"\n Constructor without parameters.. ";
num1= 0;
num2= 0;
sum = 0;
```

நிரல் பெயர்ப்பி, கொடுக்கப்பட்ட உறுப்பு செயற்கூறின் பெயர் மற்றும் திருப்பியனுப்பும் தரவினத்தை கொண்டு, அந்த செயற்கூறு ஒரு ஆக்கி செயற்கூறு எனத் தெரிந்து கொள்கிறது

```
}
add ( int s1, int s2 )
{
cout<<"\n Parameterized constructor... ";
num1= s1;
num2=s2;
sum=0;
}
add (add &a)
{
cout<<"\n Copy Constructor ... ";
num1= a.num1;
num2=a.num2;
sum = 0;
}
void getdata()
{
cout<<"\nEnter data ... ";
cin>>num1>>num2;
}
void addition()
{
sum=num1+num2;
}
void putdata()
{
cout<<"\n The numbers are..";
cout<<num1<<'\t'<<num2;
cout<<"\n The sum of the numbers are.. "<< sum;
}
};
int main()
{
add a, b (10, 20) , c(b);
a.getdata();
a.addition();
b.addition();
c.addition();
```

```
cout<<"\n Object a : ";
a.putdata();
cout<<"\n Object b : ";
b.putdata();
cout<<"\n Object c.. ";
c.putdata();
return 0;
}
```

வெளியீடு

```
    Constructor without parameters..
    Parameterized constructor...
    Copy Constructor ...
    Enter data ...  20 30
    Object a :
        The numbers are..20    30
        The sum of the numbers are.. 50
    Object b :
        The numbers are..10    20
        The sum of the numbers are.. 30
    Object c..
        The numbers are..10    20
        The sum of the numbers are.. 30
```

குறிப்பு

ஆக்கிகள் பல இடம் பெற்றிருப்பதால், ஒரு பொருள் உருவாக்கப்படும் போதே செயலுருபுகள் அவற்றிற்கு அனுப்பி வைக்கப்பட வேண்டும்.

நிரல்15.5 ஓர் இனக்குழுவில் ஆக்கி பணிமிகுப்பைப் பயன்படுத்தி செவ்வகத்தின் சுற்றளவை கணிக்கும் ஒரு நிரல்

```
// constructor declared as outline member function
#include<iostream>
using namespace std;
class Perimeter
{
int l, b, p;
public:
Perimeter ();
Perimeter (int);
Perimeter (int,int);
```

```
Perimeter (Perimeter&);
void Calculate();
};
Perimeter :: Perimeter ()
{
cout<<"\n Enter the value of length and breath";
cin>>l>>b;
cout<<"\n\nNonParameterizd constructor ";
}
Perimeter ::Perimeter (int a)
{
l=b=a;
cout<<"\n\n Parameterizd constructor with one argument ";
}
Perimeter ::Perimeter (int l1, int b1)
{
cout<<"\n\n Parameterizd constructor with 2 argument ";
l=l1;
b=b1;
}
Perimeter ::Perimeter (Perimeter &p)
{
l= p.l;
b= p.b;
cout<<"\n\n copy constructor  ";
}
void Perimeter ::Calculate(){
p = 2*(l+b);
cout<<p;
}
int main ()
{
Perimeter Obj;
cout<<"\n perimeter of rectangle is ";
Obj. Calculate ();
Perimeter Obj1(2);
cout<<"\n perimeter of rectangle ";
Obj1.Calculate ();
Perimeter Obj2 (2, 3);
cout<<"\n perimeter of rectangle ";
Obj2.Calculate ();
Perimeter obj3 (Obj2);
cout<<"\n perimeter of rectangle ";
obj3.Calculate ();
return 0;
}
```


Chapter 15.indd 246 03/09/18 5:35 PM

வெளியீடு

```
Enter the value of length and breath 10 20
Non Parameterizd constructor
perimeter of rectangle is 60
Parameterizd constructor with one argument
perimeter of rectangle 8
Parameterizd constructor with 2 argument
perimeter of rectangle 10
copy constructor
perimeter of rectangle 10
```

15.4 செயற்குறி பணிமிகுப்பு

செயற்குறி பணிமிகுப்பு என்பது +,++,-,—,+=,-=,*.<,> போன்ற வழக்கிலுள்ள C++ செயற்குறிகளுக்கு கூடுதலான செயல்பாடுகளை வரையறுப்பதைக் குறிக்கிறது. இதுவும் ஒரு பல்லுருவாக்க செயல் எனலாம். ஏனெனில், இதில் செயற்குறி பணிமிகுக்கப்பட்டு, செயற்குறிக்கு நிரலர் விரும்புகிற பொருளை வழங்குகிறது.

எடுத்துக்காட்டாக '+' என்ற செயற்குறியை பணிமிகுப்பு முழு எண்ணிற்கு கூடுதலையும், சரம் இணைத்தல் போன்ற செயல்பாட்டினை பல்வேறு தரவினங்களுக்கு பயன்படுத்திக் கொள்ளலாம். C++ மொழியில், சில செயற்குறிகளை தவிர்த்து, அனைத்து செயற்குறிகளையும் பணி மிகுப்பு செய்ய முடியும்.

பணிமிகுப்பு செய்ய முடியாத செயற்குறிகள் பின்வருமாறு

- வரையெல்லை செயற்குறி (::)
- செயற்குறி (sizeof)
- உறுப்பு தேர்வி (member selector - .)
- உறுப்பு சுட்டல் தேர்வி (member pointer selector - *)
- மும்ம செயற்குறி (ternary operator - ?:)

செயற்குறி பணிமிகுப்பின் கட்டளையமைப்பு **(Operator Overloading Syntax)**

சிறப்புச் சொல் — பணிமிகுக்கப்பட வேண்டிய செயற்குறி

```
ReturnType classname :: Operator Operator Symbol (செயலுருபுகளின் பட்டியல்)
{
        \\ Function body
}
```


Chapter 15.indd 247 03/09/18 5:35 PM

15.4.1 செயற்குறி பணிமிகுப்பின் வரம்பெல்லைகள் (Restrictions on Operator Overloading)

செயற்குறி பணிமிகுப்பினை நடைமுறைப்படுத்தும் போது பின்வரும் வரம்பெல்லைகளைக் கவனத்தில் கொள்ள வேண்டும்.

1. ஒரு செயற்குறியின் முன்னுரிமையும், திசைமுகத்தையும் மாற்ற இயலாது.

2. புதிய செயற்குறிகளை உருவாக்க முடியாது. ஏற்கனவே இருக்கும் செயற்குறிகளை மட்டுமே பணிமிகுக்க முடியும்.

3. ஒரு செயற்குறியின் அடிப்படை செயல் முறையை மறுவரையறை செய்ய முடியாது. முழு எண்கள் கூட்டப்படும் முறையை மாற்றி அமைக்க முடியாது, ஆனால் கூடுதல் செயல்பாட்டினை அந்த செயற்குறிக்கு வழங்கலாம்.

4. பணிமிகுக்கப்பட்ட செயற்குறிகள் முன்னியல்பு செயலுருபுகளைக் கொண்டிருக்காது.

5. இரும செயற்குறிகளை பணிமிகுக்கும் போது, அச்செயற்குறியின் இடப்பக்கம் அமையும் பொருள், அது வரையறுக்கப்பட்டுள்ள இனக்குழுவின் பொருளாக இருக்க வேண்டும்.

நிரல் 15.6 '+' மற்றும் '-' குறியீடுகளைப் பயன்படுத்தி இரும செயற்குறி பணிமிகுத்தலை விளக்கும் எடுத்துக்காட்டு நிரல்

```
//Complex number addition and subtraction
#include<iostream>
using namespace std;
class complex
{
int real,img;
public:
void read()
{
cout<<"\nEnter the REAL PART : ";
cin>>real;
cout<<"\nEnter the IMAGINARY PART : ";
cin>>img;
}
complex operator +(complex c2)
{
complex c3;
c3.real=real+c2.real;
c3.img=img+c2.img;
return c3;
}
```


Chapter 15.indd 248 03/09/18 5:35 PM

```
 complex operator -(complex c2)
{
complex c3;
c3.real=real-c2.real;
c3.img=img-c2.img;
return c3;
}
void display()
 {
cout<<real<<"+"<<img<<"i";
 }
 };
int main()
 {
 complex c1,c2,c3;
intchoice,cont;
 do
 {
 {
cout<<"\t\tCOMPLEX NUMBERS\n\n1.ADDITION\n\n2.SUBTRACTION\n\n";
cout<<"\nEnter your choice : ";
cin>>choice;
if(choice==1||choice==2)
{
cout<<"\n\nEnter the First Complex Number";
c1.read();
cout<<"\n\nEnter the Second Complex Number";
c2.read();
}
switch(choice)
{
 case 1     : c3=c1+c2;  // binary + overloaded
cout<<"\n\nSUM = ";
            c3.display();
            break;
case 2     : c3=c1-c2;  // binary -overloaded
cout<<"\n\nResult = ";
            c3.display();
            break;
```

```
default     : cout<<"\n\nUndefined Choice";
 }
cout<<"\n\nDo You Want to Continue?(1-Y,0-N)";
cin>>cont;
 }while(cont==1);
 return 0;
 }
```

வெளியீடு

```
        COMPLEX NUMBERS
        1.ADDITION
        2.SUBTRACTION
                Enter your choice : 1
                Enter the First Complex Number
                Enter the REAL PART : 3
                Enter the IMAGINARY PART : 4
                Enter the Second Complex Number
                Enter the REAL PART : 5
                Enter the IMAGINARY PART : 8
                SUM = 8+12i
                Do You Want to Continue?(1-Y,0-N)1
        COMPLEX NUMBERS
        1.ADDITION
        2.SUBTRACTION
                Enter your choice : 2
                Enter the First Complex Number
                Enter the REAL PART : 8
                Enter the IMAGINARY PART : 10
                Enter the Second Complex Number
                Enter the REAL PART : 4
                Enter the IMAGINARY PART : 5
                Result = 4+5i
        Do You Want to Continue?(1-Y,0-N)0
```


Chapter 15.indd 250 03/09/18 5:35 PM

நிரல் 15.7 செயற்குறி பணிமிகுப்பினைப் பயன்படுத்தி சரங்களை இணைக்கும் எடுத்துக்காட்டு நிரல்

```
#include<string.h>
#include<iostream>
using namespace std;
class strings
{
        public:
        char s[20];
        void getstring(char str[])
{
        strcpy(s,str);
  }
        void operator+(strings);
};
void strings::operator+(strings ob)
{
        strcat(s,ob.s);
        cout<<"\nConcatnated String is:"<<s;
}
int main()
{
        strings ob1, ob2;
        char string1[10], string2[10];
        cout<<"\nEnter First String:";
        cin>>string1;
        ob1.getstring(string1);
        cout<<"\nEnter Second String:";
        cin>>string2;
        ob2.getstring(string2);
        //Calling + operator to Join/Concatenate strings
        ob1+ob2;
        return 0;
}
```

வெளியீடு

```
Enter First String:COMPUTER
Enter Second String:SCIENCE
Concatenated String is:COMPUTERSCIENCE
```


Chapter 15.indd 251 03/09/18 5:35 PM

நினைவில் கொள்க:

- C++ மொழியில் பல்லுருவாக்கம் செயற்கூறு பணிமிகுப்பு மற்றும் செயற்குறி பணிமிகுப்பு ஆகியவற்றின் மூலம் நிறைவேற்றப்படுகிறது.
- பணிமிகுப்பு என்பது ஒரே பெயர், ஒன்றுக்கு மேற்பட்ட பல்வேறு பொருளைக் குறிக்கிறது.
- பணிமிகுத்த செயற்கூறு என்பது ஒன்றுக்கு மேற்பட்ட தனித்த பொருள் கொண்ட செயற்கூறினை குறிக்கிறது.
- பணிமிகுத்த செயற்கூறுகள் ஒரே பெயரைக் கொண்டிருக்கும். ஆனால் முன்வடிவு வேறுபட்டிருக்கும் (அளபுருக்களின் எண்ணிக்கை மற்றும் தரவினம்)
- செயற்கூறின் அளபுருக்களின் பட்டியல் செயற்கூறு முன்வடிவு எனப்படுகிறது.
- இரண்டு செயற்கூறுகளில் ஒன்று குறிப்பு அளபுருக்களையும் மற்றொன்று மதிப்பு அளபுருக்களையும் கொண்டிருந்தால் அவற்றை பணிமிகுக்க முடியாது.
- சாதாரண செயற்கூறுகளைப் போல் உறுப்பு செயற்கூறுகளையும் பணிமிகுக்க முடியும்.
- ஓர் இனக்குழு பணிமிகுக்கப்பட்ட ஆக்கிகளை கொண்டிருக்க முடியும். அழிப்பிகள் பணிமிகுக்கப்பட முடியாது.
- ஒரு செயற்குறிக்குப் புதிய பொருளை வழங்கும் செயல்நுட்பமே செயற்குறி பணிமிகுப்பு என்றழைக்கப்படுகிறது.
- செயற்குறி பணிமிகுப்பு C++ செயற்குறிகள் பலவற்றிற்கு புதிய வரையறுப்புகளை அளிக்கிறது.
- பயனர் வரையறுக்கும் இனங்களையும் பொருள்களைப் பணிமிகுக்க முடியும்.
- பணிமிகுக்கப்பட்ட செயற்குறியின் வரையறை இவ்வாறு அமையும். சிறப்பு சொல் “Operator” அடுத்து செயற்குறி குறியீடு அமைந்திருக்கும்.
- கீழ்கண்ட C++ செயற்குறிகளை தவிர்த்து, அனைத்து செயற்குறிகளையும் பணிமிகுக்க முடியும்.
- இனக்குழு உறுப்பு அணுகல் செயற்குறி (., .*) ,
- வரையெல்லை செயற்குறி (::),
- அளவு செயற்குறி (sizeof)
- நிபந்தனை செயற்குறி (?:)

மதிப்பீடு

N29WHE

பகுதி அ

I சரியான விடை யை தேர்வு செய்யவும் :

1. பின்வருவனவற்றுள் எது செயற்கூறுகளுக்கு வேறுபட்ட பொருள் உள்ளதை குறிக்கிறது?

   (அ) செயற்கூறு பணிமிகுப்பு (ஆ) உறுப்பு பணிமிகுப்பு

   (இ) செயற்குறி பணிமிகுப்பு (ஈ) செயற்பாடு பணிமிகுப்பு

2. பின்வருவனவற்றுள், எது நிரலின் ஒப்பீடுகளின் எண்ணிக்கையை குறைக்கிறது ?

   (அ) செயற்குறி பணிமிகுப்பு (ஆ) செயற்பாடு பணிமிகுப்பு

   (இ) செயற்குறி பணிமிகுப்பு (ஈ) உறுப்பு பணிமிகுப்பு

3. $ என்ற குறியீட்டை 10 முறை வெளியிட கீழ்காணும் நிரலில் dispchar() என்ற செயற்கூறை எவ்வாறு அழைப்பாய்?

Chapter 15.indd 252 03/09/18 5:35 PM

```
void dispchar(char ch='$',int size=10)
{
for(int i=1;i<=size;i++)
cout<<ch;
}
```

(அ) dispchar(); (ஆ) dispchar(ch,size);

(இ) dispchar($,10); (ஈ)dispchar('$',10 times);

4. பின்வருவனவற்றுள் செயற்கூறு பணிமிகுப்பு சார்ந்த எந்த கூற்று சரி கிடையாது?

(அ) பணிமிகுக்கப்பட்ட செயற்கூறுகள் முன்வடிவில் வேறுபட்டு இருக்க வேண்டும்.

(ஆ) செயற்கூறு பணிமிகுப்பின் போது திருப்பி அனுப்பும் தரவினமும் கருத்தில் கொள்ள வேண்டும்.

(இ) பணிமிகுக்கப்பட்ட செயற்கூறின் முன்னியல்பு அளபுருக்கள் பணிமிகுக்கப்படும் போது கருத்தில் கொள்ளப்படுவதில்லை.

(ஈ) அழிப்பி செயற்கூறுகள் பணிமிகுக்கப்பட முடியாது.

5. பின்வருவனவற்றுள் எது பிழையான செயற்கூறு பணிமிகுப்பு முன்வடிவாகும்?

(அ) Void fun (int x);
Void fun (char ch) ;

(ஆ) Void fun (int x);
Void fun (int y);

(இ) Void fun (double d);
Void fun (char ch);

(ஈ) Void fun (double d);
Void fun (int y);

6. கொடுக்கப்பட்ட சிறு நிரலில் பின்வரும் எந்த செயற்கூறு சேர்க்கை, செயற்கூறு பணிமிகுப்பாக எடுத்துக் கொள்ளப்படாது ?

```
voidprint(char A,int B); // F1
voidprintprint(int A, float B); // F2
voidPrint(int P=10); // F 3
voidprint(); // F4
```

(அ) F1,F2,F3,F4 (ஆ) F1,F2,F3

(இ) F1,F2,F4 (ஈ) F1,F3,F4

7. பின்வரும் எந்த செயற்குறியை நிரல்பெயர்ப்பி தானமைவாக பணிமிகுக்கும்?

(அ) * (ஆ) + (இ) += (ஈ) ==

பின்வரும் நிரலில் அடிப்படையில், (8) முதல் (10) வரை உள்ள வினாக்களுக்கு விடையளி

```
#include<iostream>
using namespace std;
```


Chapter 15.indd 253 03/09/18 5:35 PM

```
class Point {
private:
int x, y;
public:
Point(int x1,int y1)
 {
 x=x1;y=y1;
 }
void operator+(Point &pt3);
void show() {cout << "x = " << x << ", y = " << y; }
};
void Point::operator+(Point &pt3)
{
x += pt3.x;
y += pt3.y;
}
int main()
{
Point pt1(3,2),pt2(5,4);
pt1+pt2;
pt1.show();
return 0;
}
```

8. பின்வரும் எந்த செயற்குறி பணி மிகுக்கப்பட்டுள்ளது?

(அ) + (ஆ) operator

(இ) :: (ஈ) =

9. பின்வரும் எந்த கூற்று செயற்குறி பணிமிகுப்பைச் செயல்படுத்தும்?

(அ) pt1+pt2; (ஆ) Point pt1(3,2),pt2(5,4);

(இ) pt1.show(); (ஈ) return 0;

10. மேற்கண்ட நிரலின் வெளியீடு என்னவாக இருக்கும்?

(அ) x=8, y=6 (ஆ) x=14,y=14

(இ) x=8 (ஈ) = x=5, y=9 y=6

பகுதி– ஆ

II குறுவினாக்கள் (2 மதிப்பெண்கள்)

1. செயற்கூறு பணிமிகுப்பு என்றால் என்ன?
2. பணிமிகுக்க முடியாத செயற்குறிகளைப் பட்டியலிடுக.
3. class add{int x; public: add(int)}; இனக்குழுவின் வெளியே ஆக்கி வரையறுப்பை எழுதுக.
4. ஒரு செயற்கூறின் திருப்பி அனுப்பும் தரவினம் செயற்கூறு பணிமிகுப்பிற்கு உதவுமா?
5. ஒரு செயற்கூறினைப் பணிமிகுத்தலால் பயன் யாது?

பகுதி– இ

III சிறுவினாக்கள் (3 மதிப்பெண்கள்)

1. செயற்கூறு பணிமிகுப்பிற்கான விதிமுறைகள் யாவை?
2. பல செயற்கூறுகள் இருக்கும் போது, நிரல் பெயர்ப்பி அவற்றுள் எந்த செயற்கூறினை செயல்படுத்த வேண்டும் என்பதை எப்படி தீர்மானிக்கும் ? எ.கா.தருக.
3. செயற்குறி பணிமிகுப்பு என்றால் என்ன? பணிமிகுப்பு செய்யக்கூடிய செயற்குறிகளுள் சிலவற்றை கூறு.
4. ஆக்கியை பணிமிகுத்தலால் விளையும் நன்மைகளைப் பற்றி விவாதிக்கவும்?
5. class sale (int cost, discount ;public: sale(sale &); குறிப்பிட்ட செயற்கூறினுக்கு ஒரு inline அல்லாத வரையறுத்தலை எழுது.

பகுதி – ஈ

IV பெருவினாக்கள் (5 மதிப்பெண்கள்)

1. செயற்குறி பணிமிகுப்பிற்கான விதிமுறைகள் யாவை?
2. பின்வரும் இனக்குழு நிரலைப் படித்துப் பார்த்து, (i) முதல் (v) வரையிலான வினாக்களுக்கு விடையளி

```
class Book
{
int BookCode ; char Bookname[20];float fees;
public:
Book( )                    // செயற்கூறு 1
{
```


Chapter 15.indd 255 03/09/18 5:35 PM

```
fees=1000;
BookCode=1;
strcpy   (Bookname,"C++");
}
void display(float C)      // செயற்கூறு 2
{
cout<<BookCode<<":"<<Bookname<<":"<<fees<<endl;
}
~Book( )                   // செயற்கூறு 3
{
cout<<"End of Book Object"<<endl;
}
Book (intSC,char S[ ],float F)  ;          // செயற்கூறு 4
};
```

(i) மேற்கூறிய நிரலில், செயற்கூறு 1 மற்றும் செயற்கூறு 4 என்ற செயற்கூறுகளை ஒன்று சேர்த்து எவ்வாறு குறிப்பிடலாம்?

(ii) செயற்கூறு 3 எந்த கருத்துருக்களை விளக்குகிறது? இந்த செயற்கூறு எப்பொழுது அழைக்கப்படும்/செயல்படுத்தப்படும்.

(iii) செயற்கூறு 3 பயன் யாது?

(iv) செயற்கூறு 1 மற்றும் செயற்கூறு 2 ஆகிய செயற்கூறுகளை அழைக்கும் கூற்றுகளை main () செயற்கூறில் எழுதுக.

(v) செயற்கூறு 4 க்கான வரையறையை எழுதுக .

3. பின்வரும் நிரலுக்கான வெளியீட்டை எழுதுக

```
#include<iostream>
using namespace std;
class Seminar
{
int Time;
public:
Seminar()
 {
Time=30;cout<<"Seminar starts now"<<endl;
```


Chapter 15.indd 256 03/09/18 5:35 PM

```
}
void Lecture()
{
cout<<"Lectures in the seminar on"<<endl;
}
Seminar(int Duration)
{
Time=Duration;cout<<"Welcome to Seminar "<<endl;
}
Seminar(Seminar &D)
{
Time=D.Time;cout<<"Recap of Previous Seminar Content "<<endl;
}
~Seminar()
{
cout<<"Vote of thanks"<<endl;
}
};
int main()
{
Seminar s1,s2(2),s3(s2);
s1.Lecture();
return 0;
}
```

4. **பின்வரும் நிரலில் உள்ள பிழைகளை கண்டறிந்து திருத்தவும்.**

```
#include<iostream>
using namespace std;
class String
{
public:
charstr[20];
```

```
public:
voidaccept_string
{
cout<<"\n Enter String          :  ";
cin>>str;
}
display_string()
{
cout<<str;
}
String operator *(String x)  //Concatenating String
{
String s;
strcat(str,str);
strcpy(s.str,str);
goto s;
}
}
int main()
{
String str1, str2, str3;
str1.accept_string();
str2.accept_string();
cout<<"\n\n First String is        :  ";
str1=display_string();
cout<<"\n\n Second String is       :  ";
str2.display_string();
str3=str1+str2;
cout>>"\n\n Concatenated String is   :  ";
str3.display_string();
return 0;
```

```
}
```

5. பின்வரும் நிரலின் அடிப்படையில் வினாக்களுக்கு விடையளி:

```
#include<iostream>
#include<string.h>
using namespace std;
class comp {
public:
chars[10];
voidgetstring(char str[10])
{
strcpy(s,str);
}
void operator==(comp);
};
void comp::operator==(comp ob)
{
if (strcmp(s,ob.s)==0)
cout<<"\nStrings are Equal";
else
cout<<"\nStrings are not Equal";
}
intmain()
{
compob, ob1;
char string1[10], string2[10];
cout<<"Enter First String:";
cin>>string1;
ob.getstring(string1);
cout<<"\nEnter Second String:";
cin>>string2;
ob1.getstring(string2);
```


Chapter 15.indd 259 03/09/18 5:35 PM

```
ob==ob1;
return 0;
}
```

(i) நிரலின் இறுதி வரை நீடித்திருக்கும் பொருள்களை கூறு.

(ii) நிரலின் இயக்கத்திற்கிடையே அழிந்து விடும் பொருளை கூறு.

(iii) பணிமிகுக்கப்பட்ட செயற்குறி மற்றும் அதனை அழைக்க பயன்படும் கூற்றினை எழுதுக.

(iv) பணிமிகுப்பு செய்யப்பட்ட உறுப்பு செயற்கூறின் முன்வடிவை எழுதுக.

(v) பணிமிகுக்கப்பட்ட செயற்குறிக்கு எந்த வகையான செயலேற்பிகள் பயன்படுத்தப்பட்டுள்ளன?

(vi) எந்த ஆக்கி செயல்படுத்தப்படும்? நிரலின் வெளியீட்டை எழுது.

ஆய்வறிக்கை

தொடக்கத்தில் உன் சிறுசேமிப்பில் தொகை ரூ.500 உள்ளது என்று வைத்துக்கொள்வோம். அதில் மேலும் சில தொகையினை சேமிக்க விரும்புகிறாய். தொடக்க மதிப்பு ரூ.500 கொண்ட amount என்னும் தரவு உறுப்பைக் கொண்டு deposit என்ற இனக்குழுவை உருவாக்கு. அதில், பின்வரும் மூன்று ஆக்கிகளை வரையறுக்கவும்.

1 – அளபுருக்கள் இல்லாமல் – (எந்த தொகையும் சிறுசேமிப்பில் சேர்க்கப்பட மாட்டாது).

2 – சேமிப்பில் சேர்க்கப்பட வேண்டிய தொகையினை கொண்ட அளபுருக்கள்

3.ஒவ்வொரு முறையும் தொகையை சேர்க்கும் போது அதற்கு சமமான தொகையும் தாமாகவே சேர்த்துக் கொள்ளப்பட வேண்டும்.

Deposit இனக்குழுவிற்கு ஒரு பொருளை உருவாக்கி, சேமிப்பில் உள்ள மொத்த தொகையினை வெளியிடு.

சொற்களஞ்சியம்

முன்வடிவு	அளபுருக்களின் எண்ணிக்கை மற்றும் தரவினம்
பல்லுருவாக்கம்	பல வடிவங்கள்
முன்னியல்பு செயலுருபு	தொடக்க மதிப்பிருத்தல்

துணை நூல்கள்:

1. Object Oriented Programming with C++ (4th Edition), Dr. E. Balagurusamy, Mc.Graw Hills.
2. The Complete Reference C++ (Forth Edition), Herbert Schildt. Mc.Graw Hills.
3. Computer Science with C++ (A text book of CBSE XI and XII), SumitaArora, DhanpatRai& Co.
4. A text book of CBSE XI and XII computer science by PreetiArora and Pinky Gupta.
5. Computer Science with C++ Reeta shoo and Gagansahoo
6. The C++ Programming Language,BjarneStroustrup
7. C++ Primer (5th Edition) by S. B. Lippman, J. Lajoie

பாடம் 16

N2DJ6B

மரபுரிமம் (INHERITANCE)

கற்றலின் நோக்கங்கள்

இந்த பாடப்பகுதியை கற்ற பின் மாணவர் அறிந்துகொள்வது.

- மரபுரிமத்தின் நோக்கத்தைப் புரிந்துகொள்ளுதல்.
- மரபுரிமத்தை பயன்படுத்தி C++ நிரல்களை உருவாக்குதல்.
- மரபுரிமத்தின் கருத்துருக்களைக் கொண்டு நிரல்களை இயக்குதல் மற்றும் பிழைத் திருத்தம் செய்தல்.

16.1. மரபுரிமம் - ஓர் அறிமுகம்.

ஒரு பொருள் நோக்கு நிரலாக்கத்தின் மிக முக்கிய பண்புகூறுகளில் மரபுரிமம் ஒன்றாகும். பொருள் நோக்கு நிரலாக்கத்தில் மரபுரிமம், அடிப்படை இனக்குழுக்களின் பண்புகூறுகளை ஈட்டிக் கொள்ள புதிய இனக்குழுவிற்கும், அதனுடைய பொருள்களுக்கும் வழிவகை செய்கிறது.

மரபுரிமத்தினை செயல்படுத்த அடிப்படையாக உள்ள இனக்குழுவை, மீ இனக்குழு அல்லது அடிப்படை இனக்குழு என்கிறோம். மீ இனக்குழுவிலிருந்து தருவிக்கப்படும் இனக்குழு துணை இனக்குழு அல்லது தருவிக்கப்பட்ட இனக்குழு எனப்படுகிறது.

16.2. மரபுரிமத்தின் தேவை

பொருள் நோக்கு நிரலாக்கத்தின் முக்கிய பண்புக்கூறான மரபுரிமம், நிரல் குறிமுறையின் மறு பயனாக்கத்திற்கு வழி செய்கிறது. மரபுரிமம் என்பது ஏற்கனவே இருக்கும் இனக்குழுக்களின் அடிப்படையில் (அடிப்படை இனக்குழு) புதிய இனக்குழுக்களை தருவிக்கும் (தருவிக்கப்பட்ட இனக்குழுக்கள்) செயல்முறை ஆகும். மரபுரிமம் ஒரு இனக்குழுவிலிருந்து மற்றொரு இனக்குழுவிற்கு அனைத்து குறிமுறைகளையும் (Private என வகைப்படுத்தப்பட்ட தரவு உறுப்புகள், உறுப்பு செயற்கூறுகள் தவிர்த்து) தருவித்துக் கொள்ள அனுமதிக்கிறது. தருவிக்கப்பட வேண்டிய இனக்குழுவை அடிப்படை இனக்குழு அல்லது தாய் இனக்குழு (parent class) என்றும் ஒரு இனக்குழுவிலிருந்து மரபுரிமையாக பெறப்பட்ட இனக்குழுவை தருவிக்கப்பட்ட இனக்குழு அல்லது சேய் இனக்குழு என்று கூறுகிறோம். தருவிக்கப்பட்ட இனக்குழுக்கள் சக்திமிக்கவை. இது கூடுதல் பண்புக்கூறுகளையும், செயல்முறைகளையும் பெற்றுக்கொண்டு செயல்திறனை அதிகரிக்க செய்கிறது.

குறிப்பு

மரபுரிமத்தின் நன்மைகளாவன:

- இது நடைமுறை வாழ்வில் உள்ள உறவு நிலையை சிறப்பாக எடுத்துக்காட்ட உதவுகிறது.
- நிரல் குறிமுறையின் மறுபயனாக்கத்தை சாத்தியமாக்கி உள்ளது.
- இது மாற்றங்களை ஆதரிக்கும்.

16.3. மரபுரிமத்தின் வகைகள்

மரபுரிமத்தில் பல வகைகள் உள்ளன. ஒரு வழி மரபுரிமம், பலவழி மரபுரிமம், பல நிலை மரபுரிமம், கலப்பு மரபுரிமம் மற்றும் படிமுறை மரபுரிமம்.

1. ஒரு வழி மரபுரிமம்

ஒரேயொரு இனக்குழுவை அடிப்படையாகக் கொண்டு தருவிக்கப்பட்ட இனக்குழுவை உருவாக்குவது ஒரு வழி மரபுரிமம் ஆகும்.

2. பலவழி மரபுரிமம்

பல அடிப்படை இனக்குழுக்களிலிருந்து தருவிக்கப்பட்ட இனக்குழுவை உருவாக்குவது பல வழி மரபுரிமம் ஆகும்.

3. படிமுறை மரபுரிமம்

ஒன்றுக்கு மேற்பட்ட தருவிக்கப்பட்ட இனக்குழுக்கள் ஒரு அடிப்படை இனக்குழுவிலிருந்து தருவிக்கப்படுமாயின் அது படிமுறை மரபுரிமம் எனப்படும்.

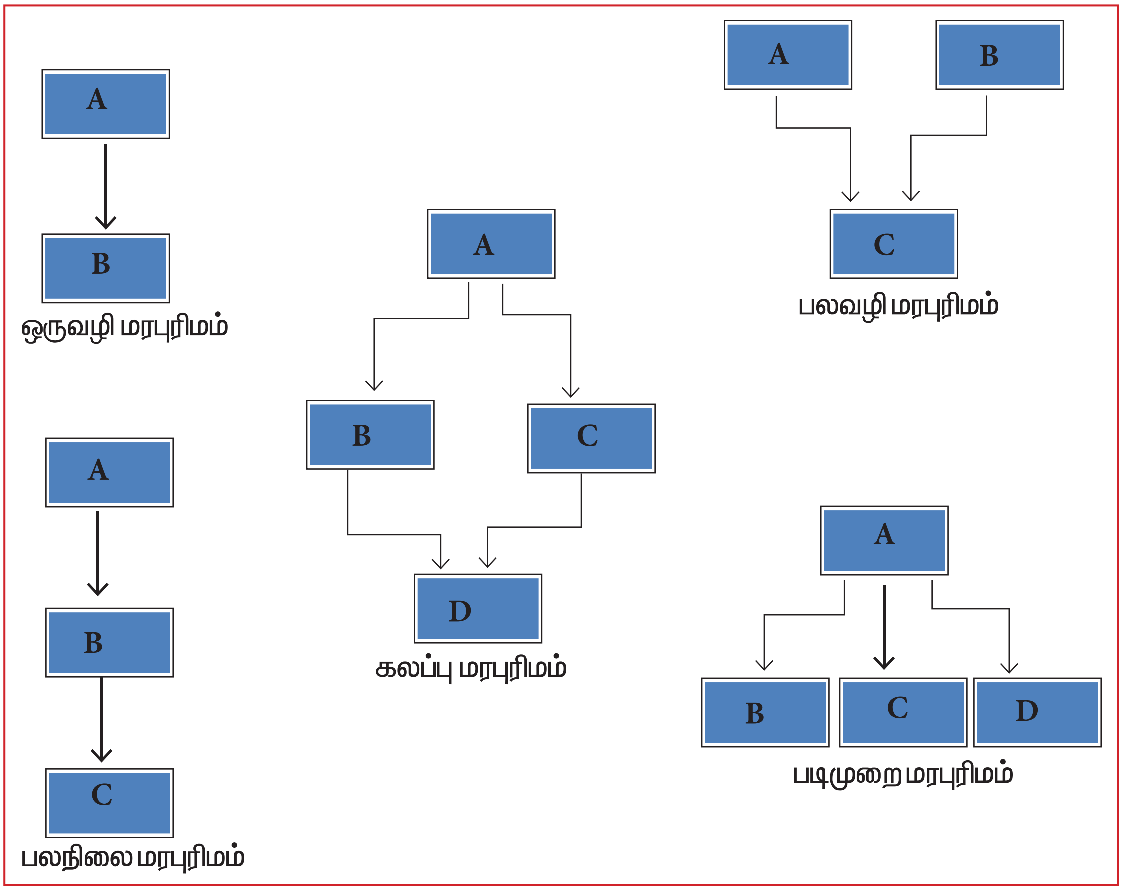
4. பலநிலை மரபுரிமம்

மரபுரிமத்தின் மாறும் இயல்புடைய பண்புகள் இந்த வகை மரபுரிமத்தில் பிரதிபலிக்கின்றன. ஓர் இனக்குழு தருவிக்கப்பட்ட இனக்குழுவைக்கொண்டு தருவிக்கப்பட்டால், அது பலநிலை மரபுரிமம் எனப்படும்.

5. கலப்பு மரபுரிமம்.

ஒன்றுக்கு மேற்பட்ட மரபுரிம வகைகளை இணைப்பதன் மூலம் கலப்பு மரபுரிம வகையை உருவாக்கலாம். இது, பலநிலை அல்லது பலவழி மரபுரிமம், படிமுறை மற்றும் பலநிலை மரபுரிமம், அல்லது படிமுறை, பலநிலை மற்றும் பல வகை கலப்பினமாக இருக்கலாம்.

மரபுரிமத்தின் பல்வேறு வகைகளை விளக்கும் படம் கீழே கொடுக்கப்பட்டுள்ளன.



16.4. தருவிக்கப்பட்ட இனக்குழு மற்றும் அடிப்படை இனக்குழு

தருவிக்கப்பட்ட இனக்குழுவை வரையறுக்கும் போது, தருவிக்கப்பட்ட இனக்குழு அது எந்த இனக்குழுவிலிருந்து தருவிக்கப்படுகிறதோ, அதனைக் குறிப்பிட வேண்டும்.

தருவிக்கப்பட்ட இனக்குழுவை வரையறுக்கும் போது கீழ்காணும் குறிப்புகளைக் கவனத்தில் கொள்ள வேண்டும்.

அ) Class என்னும் சிறப்புச் சொல் இடம்பெற வேண்டும்.

ஆ) Class என்ற சொல்லை அடுத்து, தருவிக்கப்படும் இனக்குழுவின் பெயர் இடம் பெற வேண்டும்

இ) ஒற்றை முக்காற்புள்ளி (:) இடம் பெற வேண்டும்.

ஈ) Private, public அல்லது protected ஆகியவற்றுள் எத்தகைய அணுகியல்புடன் (காண்புநிலை பாங்கு) தருவிக்கப்படுகிறது என குறிப்பிட வேண்டும். காண்புநிலை பாங்கு எதுவும் குறிப்பிடப்படவில்லையெனில், தானமைவாக காண்புநிலை private எனக் கொள்ளப்படும்.

உ) ஒன்றுக்கு மேற்பட்ட அடிப்படை இனக்குழுக்கள் (தாய் இனக்குழுக்கள்) இருப்பின், அவற்றை காற்புள்ளியிட்டு பிரிக்க வேண்டும்.

கட்டளையமைப்பு

class தருவிக்கப்பட்ட_இனக்குழு_பெயர்: அணுகியல்பு வரையறுப்பி அடிப்படை_இனக்குழுவின்_பெயர்

{

// தருவிக்கப்பட்ட இனக்குழுவின் உறுப்புகள்

};

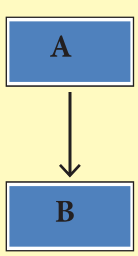
16.4.1 ஒருவழி மரபுரிமம் (Single Inheritance)

பல்வேறு வகையான மரபுரிமத்தை விளக்கும் எடுத்துக்காட்டு கீழே கொடுக்கப்பட்டுள்ளது.

எடுத்துக்காட்டு: 16.1 ஒரு வழி மரபுரிமம்

```
# include <iostream>
using namespace std;
class student        //base class
{
private :
char name[20];
int rno;
public:
void acceptname()
{
cout<<"\n Enter roll no and name .. ";
cin>>rno>>name;
}
```

தருவிக்கப்பட்ட இனக்குழு அடிப்படை இனக்குழுவின் அனைத்து உறுப்புகளையும் மரபுரிமமாக பெற்றிருந்தாலும், private அல்லாத உறுப்புகளை மட்டுமே அணுக முடியும்.



ஒருவழி மரபுரிமம்

Chapter 16.indd 263 03/09/18 5:51 PM

```
void displayname()
{
cout<<"\n Roll no :-"<<rno;
cout<<"\n Name :-"<<name<<endl;
}
};
class exam : public student          //derived class with single base class
 {
public:
          int  mark1, mark2 ,mark3,mark4,mark5,mark6,total;
  void acceptmark()
  {
    cout<<"\n Enter lang,eng,phy,che,csc,mat marks.. ";
    cin>>mark1>>mark2>>mark3>>mark4>>mark5>>mark6;
}
void displaymark()
{
cout<<"\n\t\t Marks Obtained ";
cout<<"\n Language.. "<<mark1;
cout<<"\n English .. "<<mark2;
cout<<"\n Physics .. "<<mark3;
cout<<"\n Chemistry.. "<<mark4;
cout<<"\n Comp.sci.. "<<mark5;
cout<<"\n Maths ..   "<<mark6;
}
 };
int main()
{
  exam e1;
 e1.acceptname();      //calling base class function using derived class object
 e1.acceptmark();
 e1.displayname();       //calling base class function using derived class object
 e1.displaymark();
 return 0;
}
```

வெளியீடு:

```
Enter roll no and name .. 1201 KANNAN
Enter lang,eng,phy,che,csc,mat marks.. 100 100 100 100 100 100
Roll no :-1201
Name :-KANNAN
Marks Obtained
Language.. 100
English .. 100
Physics .. 100
Chemistry.. 100
Comp.sci.. 100
Maths ..   100
```

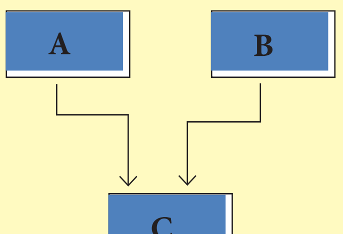
மேற்கண்ட நிரலில் **"exam"** என்ற தருவிக்கப்பட்ட இனக்குழு **"student"** என்ற அடிப்படை இனக்குழுவின் அனைத்து உறுப்புகளையும் மரபுவழி பெற்றுக் கொள்கிறது. ஆனால், **private** அல்லாத உறுப்புகளை மட்டும் அணுகும் உரிமையை பெறுகிறது.

16.4.2 பலவழி மரபுரிமம் (Multiple Inheritance)

எடுத்துக்காட்டு நிரல் 16.2 பல வழி மரபுரிமம்

```
# include <iostream>
using namespace std;
class student            //base class
{
private :
char name[20];
int rno;
public:
void acceptname()
{
cout<<"\n Enter roll no and name .. ";
cin>>rno>>name;
}
void displayname()
{
cout<<"\n Roll no :-"<<rno;
cout<<"\n Name :-"<<name<<endl;
```

தருவிக்கப்பட்ட இனக்குழு அடிப்படை இனக்குழுவை இடமிருந்து வலம் வரிசை முறையில் மரபுரிமத்தை பெற்று கொள்கிறது. .



பலவழி மரபுரிமம்

```
;
}
};
class detail                    //Base class
{
            int dd,mm,yy;
            char cl[4];
public:
void acceptdob()
{
cout<<"\n Enter date,month,year in digits and class .. ";
cin>>dd>>mm>>yy>>cl;
}
void displaydob()
{
cout<<"\n class:-"<<cl;
cout<<"\t\t DOB     :  "<<dd<<" - "<<mm<<" –" <<yy<<endl;
}
};
class exam : public student,public detail          //derived class with multiple  base
class
 {
    public:
int  mark1, mark2 ,mark3,mark4,mark5,mark6,total;
  void acceptmark()
  {
    cout<<"\n Enter lang,eng,phy,che,csc,mat marks.. ";
    cin>>mark1>>mark2>>mark3>>mark4>>mark5>>mark6;}
void displaymark()
{
cout<<"\n\t\t Marks Obtained ";
cout<<"\n Language.. "<<mark1;
cout<<"\n English .. "<<mark2;
cout<<"\n Physics .. "<<mark3;
cout<<"\n Chemistry.. "<<mark4;
cout<<"\n Comp.sci.. "<<mark5;
cout<<"\n Maths ..   "<<mark6;
}
 };
```

```
 e1.acceptname();      //calling base class function using derived class object
e1.acceptdob();         //calling base class function using derived class object
 e1.acceptmark();
 e1.displayname();      //calling base class function using derived class object
e1.displaydob();        //calling base class function using derived class object
 e1.displaymark();
 return 0;
}
```

வெளியீடு:

```
        Enter roll no and name .. 1201 MEENA
        Enter date,month,year in digits and class .. 7 12 2001 XII
        Enter lang,eng,phy,che,csc,mat marks.. 96 98 100 100 100 100
        Roll no :-1201
        Name :- MEENA
        class     :-XII          DOB   :  7 - 12 -2001
        Marks Obtained
        Language.. 96
        English .. 98
        Physics .. 100
        Chemistry.. 100
        Comp.sci.. 100
        Maths ..  100
```

மேற்கண்ட நிரலில் "exam" என்ற இனக்குழு "student" மற்றும் "detail" என்ற அடிப்படை இனக்குழுக்களைக் கொண்டு தருவிக்கப்பட்டுள்ளது. ஆகையால், அது இரண்டு இனக்குழுக்களின் (அடிப்படை) உறுப்புகளை அணுக முடியும்.

குறிப்பு

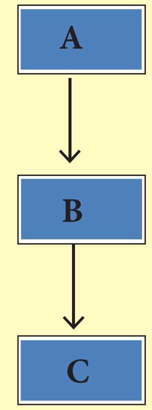
பலவழி மரபுரிமத்தில், அடிப்படை இனக்குழுக்களுக்கிடையே எந்த உறவு முறையும் இருக்காது. எனினும் தருவிக்கப்பட்ட இனக்குழு அனைத்து அடிப்படை இனக்குழுக்களின் பண்புகளை பெற்றுக்கொள்கிறது.

Chapter 16.indd 267 03/09/18 5:51 PM

16.4.3 பலநிலை மரபுரிமம் (Multilevel Inheritance)

எடுத்துக்காட்டு நிரல் 16.3 பலநிலை மரபுரிமம்

பலநிலை மரபுரிமத்தில், தருவிக்கப்பட்ட இனக்குழு மற்றொரு இனக்குழுவிற்கு அடிப்படை இனக்குழுவாக அமைகிறது.



பலநிலை மரபுரிமம்

```
# include <iostream>
using namespace std;
class student              //base class
{
private :
char name[20];
int rno;
public:
void acceptname()
{
cout<<"\n Enter roll no and name .. ";
cin>>rno>>name;
}
void displayname()
{
cout<<"\n Roll no :-"<<rno;
cout<<"\n Name :-"<<name<<endl;
}};
 class exam : public student      //derived class with
single base class
{
        public:
        int  mark1, mark2 ,mark3,mark4,mark5,mark6;
        void acceptmark()
        {
        cout<<"\n Enter lang,eng,phy,che,csc,mat marks.. ";
        cin>>mark1>>mark2>>mark3>>mark4>>mark5>>mark6;
        }
void displaymark(){
cout<<"\n\t\t Marks Obtained ";
cout<<"\n Language.. "<<mark1;
cout<<"\n English .. "<<mark2;
cout<<"\n Physics .. "<<mark3;
cout<<"\n Chemistry.. "<<mark4;
cout<<"\n Comp.sci.. "<<mark5;
cout<<"\n Maths ..   "<<mark6;
}
};
```

```
class result : public exam
{
        int total;
        public:
                void showresult()
                {
                        total=mark1+mark2+mark3+mark4+mark5+mark6;
                        cout<<"\nTOTAL MARK SCORED   : "<<total;
         }
};
int main()
{
        result r1;
r1.acceptname();      //calling base class function using derived class object
r1.acceptmark();      //calling base class function which itself is a derived
        // class  function using its derived class object
r1.displayname();     //calling base class function using derived class
                      //object
r1.displaymark();     //calling base class function which itself is a derived
                      //class  function using its derived class object
r1.showresult();      //calling the child class function
return 0;
}
```

வெளியீடு:

```
        Enter roll no and name .. 1201 SARATHI
         Enter lang,eng,phy,che,csc,mat marks.. 96 98 100 100 100 100
        Roll no :-1201
        Name :-SARATHI
        Marks Obtained
        Language.. 96
        English ..  98
        Physics ..  100
        Chemistry.. 100
        Comp.sci..  100
        Maths ..    100
        TOTAL MARK SCORED    :  594
```

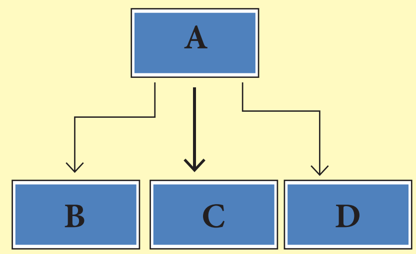
மேற்கண்ட நிரலில் "result " என்ற இனக்குழு student என்ற இனக்குழுவிலிருந்து தருவிக்கப்பட்ட இனக்குழுவான , "exam" என்ற அடிப்படை இனக்குழுவிலிருந்து தருவிக்கப்பட்டுள்ளது.

Chapter 16.indd 269 03/09/18 5:51 PM

குறிப்பு

பலநிலை மரபுரிமத்தில், உறவு நிலையின் அடிப்படையில் எத்தனை நிலை வேண்டுமென்றாலும் நீட்டித்து கொள்ளலாம். இந்த பலநிலை மரபுரிமம் வகை முன்னோர், தாய், சேய் உறவுமுறை போன்று அமையும்.

எடுத்துக்காட்டு நிரல் 16.4 படிமுறை மரபுரிமம் (Hierarchical inheritance)



படிமுறை மரபுரிமம்

```
# include <iostream>
using namespace std;
class student                //base class
{
private :
char name[20];
int rno;
public:
void acceptname()
{
cout<<"\n Enter roll no and name .. ";
cin>>rno>>name;
}
void displayname()
{
cout<<"\n Roll no :-"<<rno;
cout<<"\n Name :-"<<name<<endl;
}
};
 class qexam : public student   //derived class with single base class
 {
   public:
        int  mark1, mark2 ,mark3,mark4,mark5,mark6;
  void acceptmark()
   {
     cout<<"\n Enter lang,eng,phy,che,csc,mat marks for quarterly exam.. ";
     cin>>mark1>>mark2>>mark3>>mark4>>mark5>>mark6;
   }
void displaymark()
{
cout<<"\n\t\t Marks Obtained in quarterly";
cout<<"\n Language.. "<<mark1;
cout<<"\n English .. "<<mark2;
cout<<"\n Physics .. "<<mark3;
cout<<"\n Chemistry.. "<<mark4;
```


Chapter 16.indd 270 03/09/18 5:51 PM

```
cout<<"\n Comp.sci.. "<<mark5;
cout<<"\n Maths ..   "<<mark6;
}
 };
  class hexam : public student           //derived class with single base class
 {
    public:
            int  mark1, mark2 ,mark3,mark4,mark5,mark6;
   void acceptmark()
   {
     cout<<"\n Enter lang,eng,phy,che,csc,mat marks for halfyearly exam.. ";
     cin>>mark1>>mark2>>mark3>>mark4>>mark5>>mark6;
  }
void displaymark()
{
cout<<"\n\t\t Marks Obtained in Halfyearly";
cout<<"\n Language.. "<<mark1;
cout<<"\n English .. "<<mark2;
cout<<"\n Physics .. "<<mark3;
cout<<"\n Chemistry.. "<<mark4;
cout<<"\n Comp.sci.. "<<mark5;
cout<<"\n Maths ..   "<<mark6;
}
 };
int main()
{
  qexam q1;
  hexam h1;
 q1.acceptname();      //calling base class function using derived class object
 q1.acceptmark();      //calling base class function
 h1.acceptname();      //calling base class function using derived class object
 h1.displayname();    //calling base class function using derived class object
 h1.acceptmark();
 h1.displaymark();       //calling base class function using its
                         //derived class object
  return 0;
}
```

வெளியீடு

```
Enter roll no and name .. 1201 KANNAN
 Enter lang,eng,phy,che,csc,mat marks for quarterly exam..
95 96 100 98 100 99
Roll no :-1201
Name :-KANNAN
        Marks Obtained in quarterly
        Language.. 95
        English .. 96
        Physics .. 100
        Chemistry.. 98
        Comp.sci.. 100
        Maths ..   99
Enter roll no and name .. 1201 KANNAN
        Enter lang,eng,phy,che,csc,mat marks for halfyearly exam..
        96 98 100 100 100 100
        Roll no :-1201
Name :-KANNAN
        Marks Obtained in Halfyearly
        Language.. 96
        English .. 98
        Physics .. 100
        Chemistry.. 100
        Comp.sci.. 100
        Maths ..   100
```

மேற்கண்ட நிரலில் “student” என்ற இனக்குழுவிலிருந்து இரண்டு இனக்குழுக்கள் “qexam” மற்றும் “hexam” தருவிக்கப்பட்டுள்ளன. இங்கு ஒரு அடிப்படை இனக்குழுவிற்கு ஒன்றுக்கு மேற்பட்ட தருவிக்கப்பட்ட இனக்குழுக்கள் இருக்கின்றன. அதனால் இந்த நிரல், படிமுறை மரபுரிமத்தை சார்ந்தது.

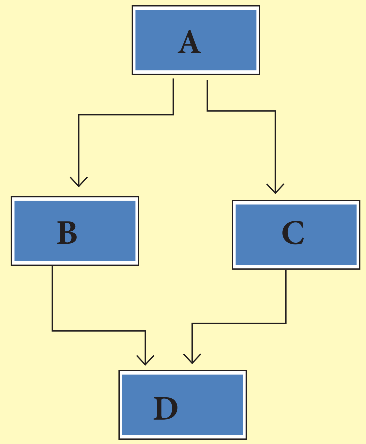
குறிப்பு

ஓர் இனக்குழு எந்த அறிவிப்புகளும் இல்லாமல் வரையறுக்கும் போது 1 பைட் அளவுள்ளதாக இருக்கும்.

class x { }; x 1 பைட் இடைத்தை எடுத்துக்கொள்ளும்

Chapter 16.indd 272 03/09/18 5:51 PM

எடுத்துக்காட்டு நிரல் 16.5 கலப்பு மரபுரிமம் (Hybrid inheritance)



கலப்பு மரபுரிமம்

```
# include <iostream>
using namespace std;
class student          //base class
{
private :
char name[20];
int rno;
public:
void acceptname()
{
cout<<"\n Enter roll no and name .. ";
cin>>rno>>name;
}
void displayname()
{
cout<<"\n Roll no :-"<<rno;
cout<<"\n Name :-"<<name<<endl;
} };
class exam : public student          //derived class with single base class
 {
        public:
        int  mark1, mark2 ,mark3,mark4,mark5,mark6;
        void acceptmark()
  {
        cout<<"\n Enter lang,eng,phy,che,csc,mat marks.. ";
        cin>>mark1>>mark2>>mark3>>mark4>>mark5>>mark6;
  }
void displaymark() {
cout<<"\n\t\t Marks Obtained ";
cout<<"\n Language.. "<<mark1;
cout<<"\n English .. "<<mark2;
cout<<"\n Physics .. "<<mark3;
cout<<"\n Chemistry.. "<<mark4;
cout<<"\n Comp.sci.. "<<mark5;
cout<<"\n Maths ..  "<<mark6;
}
 };
 class detail               //base classs 2
{
```


Chapter 16.indd 273 03/09/18 5:51 PM

```
  int dd,mm,yy;
 char cl[4];
public:
 void acceptdob()
{
cout<<"\n Enter date,month,year in digits and class .. ";
cin>>dd>>mm>>yy>>cl;
}
void displaydob()
{
cout<<"\n class      :-"<<cl;
cout<<"\t\t DOB    :  "<<dd<<" - "<<mm<<" -" <<yy<<endl;
}
};
class result : public exam,public detail   //inherits from exam ,which itself is a
                                           //derived class and also from class detail
 {
        int total;
        public:
                void showresult()
        {
                total=mark1+mark2+mark3+mark4+mark5+mark6;
                 cout<<"\nTOTAL MARK SCORED    :  "<<total;
         }
 };
int main()
{
 result r1;
 r1.acceptname();      //calling base class function using derived class object
 r1.acceptmark();      //calling base class which itsel is a derived class  function using
its derived class object
r1.acceptdob();
 cout<<"\n\n\t\t MARKS STATEMENT";
 r1.displayname();       //calling base class function using derived class object
 r1.displaydob();
r1.displaymark();         //calling base class which itsel is a derived class  function
using its derived class object
 r1.showresult();        //calling the child class function
 return 0;
}
```

வெளியீடு:

```
Enter roll no and name .. 1201 RAGU
Enter lang,eng,phy,che,csc,mat marks.. 96 98 100 100 100 100
Enter date,month,year in digits and class .. 7 12 2001 XII
MARKS STATEMENT
Roll no :-1201
Name :-RAGU
class      :-XII           DOB    : 7 - 12 -2001
Marks Obtained
Language.. 96
English .. 98
Physics .. 100
Chemistry.. 100
Comp.sci.. 100
Maths ..   100
TOTAL MARK SCORED    : 594
```

மேற்கண்ட நிரலில் **"result"** என்ற தருவிக்கப்பட்ட இனக்குழு **"student"** என்ற இனக்குழுவிலிருந்து தருவிக்கப்பட்ட இனக்குழுக்களான **"exam"** மற்றும் **"detail"** இனக்குழுவின் பண்புகளை பெற்றுக் கொள்கின்றன. இந்த மரபுரிமம் பலநிலை மற்றும் பலவழி மரபுரிமத்தை ஒருங்கே கொண்டுள்ளதால் இதனை கலப்பு மரபுரிமம் என்கிறோம்.

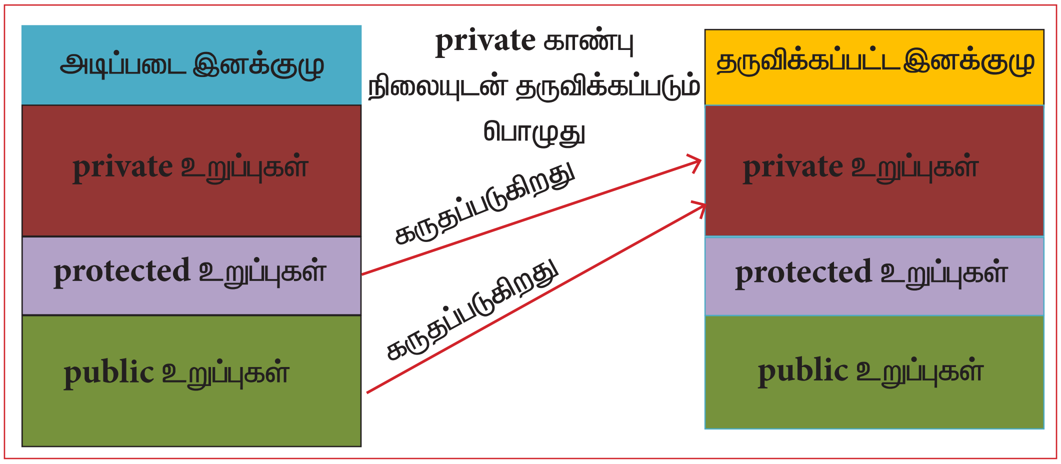
16.5 காண்புநிலை பாங்குகள் (Visibility modes)

மரபுரிமத்தின் முக்கியமான பண்புக்கூறு என்னவெனில், அடிப்படை இனக்குழுவின் எந்த உறுப்பினை தருவிக்கப்பட்ட இனக்குழு மரபுவழி பெற்றுக்கொள்ள முடியும் என்பதை தெரிந்து கொள்வதாகும். இவை காண்புநிலை பாங்குகளைக் கொண்டு நிறைவேற்றப்படுகிறது.

தருவிக்கப்பட்ட இனக்குழு அடிப்படை இனக்குழுவை அணுகும் முறையை காண்பு நிலை பாங்குகள் கட்டுப்படுத்துகிறது. **private, public, protected** என்ற மூன்று காண்புநிலை பாங்குகள் உள்ளன. முன்னியல்பான காண்புநிலை **private** ஆகும். காண்புநிலை பாங்கும், அணுகுநிலை வரையறுப்பிகளும் ஒன்று போல் இருந்தாலும், முக்கிய வேறுபாடு என்னவெனில் அணுகியல்பு வரையறுப்பிகள் இனக்குழுவிற்குள் அமைந்திருக்கும் உறுப்புகளின் அணுகுதலை கட்டுபடுத்தும் ஆனால் காண்புநிலைப் பாங்குகள் இனக்குழுவிற்குள் இருக்கும் மரபுரிமையாக பெறப்பட்ட உறுப்புகளை அணுக பயன்படுகிறது.

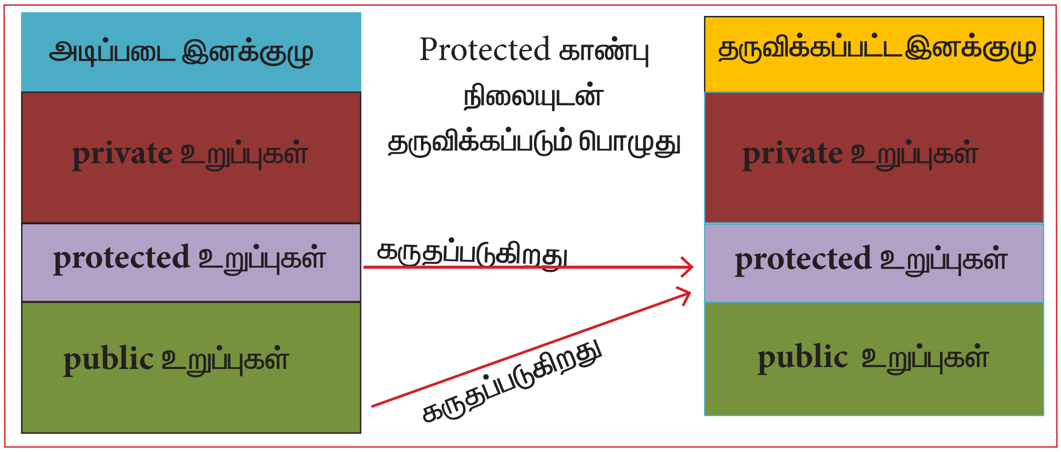
16.5.1. Private காண்புநிலை பாங்கு

ஓர் அடிப்படை இனக்குழு private என்னும் அணுகியல்புடன் தருவிக்கப்படும் போது, அடிப்படை இனக்குழுவின் public மற்றும் protected உறுப்புகள் தருவிக்கப்படும் இனக்குழுவில் private உறுப்புகளாகக் கருதப்படுகின்றன.



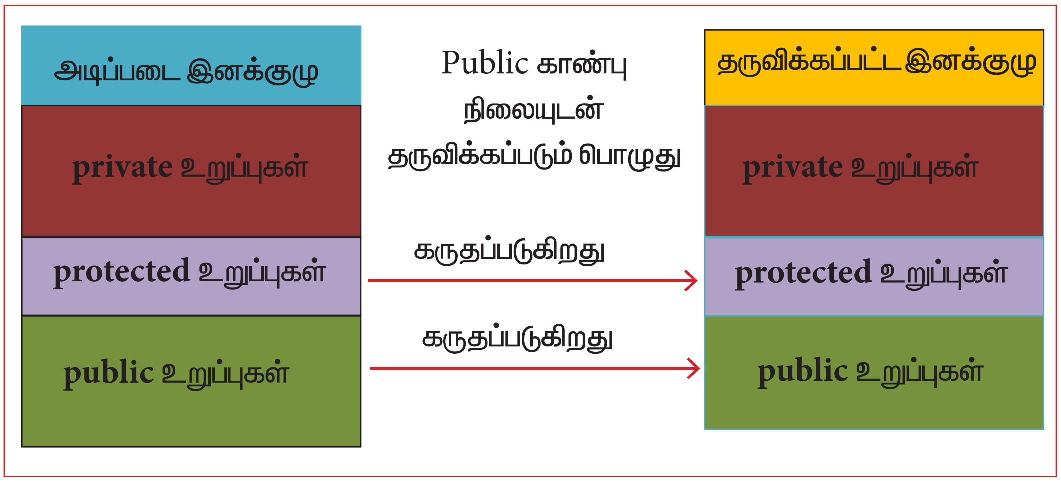
16.5.2. Protected காண்புநிலை பாங்கு

ஓர் அடிப்படை இனக்குழு protected என்னும் அணுகியல்புடன் தருவிக்கப்படும் போது, அடிப்படை இனக்குழுவின் protected மற்றும் public உறுப்புகள், தருவிக்கப்படும் இனக்குழுவில் protected உறுப்புகளாக கருதப்படுகின்றன.



16.5.3 public காண்புநிலை பாங்கு

ஓர் அடிப்படை இனக்குழு public என்னும் அணுகியல்புடன் தருவிக்கப்படும் போது, அடிப்படை இனக்குழுவின் protected உறுப்புகள், தருவிக்கப்படும் இனக்குழுவில் protected உறுப்புகளாகவும், public உறுப்புகள் public உறுப்புகளாக கருதப்படுகின்றன.



குறிப்பு

இனக்குழுக்கள் public, private, protected என எந்த வகையிலும் தருவிக்கப்பட்டாலும், அடிப்படை இனக்குழுவின் private உறுப்புகள் தருவிக்கப்படும் இனக்குழுவில் மரபுரிமையாக பெறப்படுவதில்லை. அதாவது தருவிக்கப்பட்ட இனக்குழுவில் தொடர்ந்து இருக்கம் ஆனால் அணுக முடியாது.

எடுத்துக்காட்டு நிரல் **16.6** பல்வேறு காண்புநிலை பாங்குகளை விளக்குகிறது.

```
//காண்புநிலைக் கொண்டு ஒரு வழி மரபுரிமத்தை செயல்படுத்தல்
#include <iostream>
using namespace std;
class Shape
{
  private:
        int count;
protected:
int width;
int height;
public:
voidsetWidth(int w)
{
width = w;
}
voidsetHeight(int h)
{
height = h;
}
};
class Rectangle: publicShape
{
public:
intgetArea()
{
return (width * height);
}
};
int main()
{
Rectangle Rect;
Rect.setWidth(5);
Rect.setHeight(7);
// Print the area of theobject.
cout<< "Total area: "<<Rect.getArea() <<endl;
return 0;
}
```

வெளியீடு

```
        Total area: 35
```

கீழ்கண்ட அட்டவணை, மரபுரிமம் செயல்படுத்துவதற்கு முன் இனக்குழுவினுள் வரையறுக்கப்பட்டுள்ள உறுப்புகளைக் கொண்டுள்ளது.

இனக்குழு உறுப்புகள்	காண்புநிலை பாங்குகள்		
	Private	Protected	Public
shape (அடிப்படை இனக்குழு)	int count;	int width; int height;	voidsetWidth(int) voidsetHeight(int)
Rectangle (வரையறுக்கப்பட்ட உறுப்புகளுடன் கூடிய தருவிக்கப்பட்ட இனக்குழு)			intgetArea();

கீழ்கண்ட அட்டவணை, மரபுரிமம் செயல்படுத்தப்பட்ட பின்னர் இனக்குழுவினுள் வரையறுக்கப்பட்டுள்ள உறுப்புகளை கொண்டுள்ளது.

இனக்குழு உறுப்புகள்	Public காண்புநிலை பாங்குடன் அடிப்படை இனக்குழுவின் பண்பு கூறுகளைப் பெறுதல்		
	Private	Protected	Public
shape (அடிப்படை இனக்குழு)	int count;	int width; int height;	voidsetWidth(int) voidsetHeight(int)
Rectangle (அடிப்படை இனக்குழுவின் பண்புக்கூறுகளை public காண்புநிலையுடன் தருவிக்கப்பட்ட இனக்குழு பெறுகிறது.	அடிப்படை இனக்குழுவின் private உறுப்புகளை தருவிக்கப்பட்ட இனக்குழு நேரடியாக அணுக முடியாது.	int width; int height;	intgetArea(); void setWidth(int) void setHeight(int)

protected என்ற காண்பு நிலையுடன் rectangle என்ற இனக்குழு தருவிக்கப்பட்டால், rectangle இனக்குழுவின் பண்புக்கூறுகள் கீழ்கண்ட முறையில் மாறும்.

இனக்குழு உறுப்புகள்	**protected** காண்புநிலை பாங்குடன் – (அடிப்படை இனக்குழுவின் பண்புக்கூறுகளை பெறுதல்.		
	Private	Protected	Public
shape (அடிப்படை இனக்குழு)	int count;	int width; int height;	voidsetWidth(int) voidsetHeight(int)
rectangle (அடிப்படை இனக்குழுவின் பண்புக்கூறுகளை protected காண்புநிலையுடன் தருவிக்கப்பட்ட இனக்குழு பெறுகிறது.)	அடிப்படை இனக்குழுவின் private உறுப்புகளை தருவிக்கப்பட்ட இனக்குழு நேரடியாக அணுக முடியாது.	int width; int height; void setWidth(int) void setHeight(int)	intgetArea();

Chapter 16.indd 278 03/09/18 5:51 PM

private என்ற காண்பு நிலை பாங்குடன் rectangle என்ற இனக்குழு தருவிக்கப்பட்டால், rectangle இனக்குழுவின் பண்புக்கூறுகள் கீழ்கண்ட முறையில் மாறும்.

இனக்குழு உறுப்புகள்	Private காண்புநிலை பாங்குடன் – (அடிப்படை இனக்குழுவின் பண்புக்கூறுகளை பெறுதல்.		
	Private	Protected	Public
shape (அடிப்படை இனக்குழு)	int count;	int width; int height;	voidsetWidth(int) voidsetHeight(int)
rectangle (அடிப்படை இனக்குழுவின் பண்புக்கூறுகளை private காண்புநிலையுடன் தருவிக்கப்பட்ட இனக்குழு பெறுகிறது.)	அடிப்படை இனக்குழுவின் private உறுப்புகளை தருவிக்கப்பட்ட இனக்குழு நேரடியாக அணுக முடியாது.	int width; int height; void setWidth(int) void setHeight(int)	intgetArea();

ஏற்கனவே இருக்கும் இனக்குழுவின் அடிப்படையில் புதிய இனக்குழுவை தருவிக்கும் போது, காண்பு நிலை பாங்கின் அடிப்படையில் தருவிக்கப்பட்ட இனக்குழு அடிப்படை இனக்குழுவின் பண்புகளை மரபு வழி பெறுகிறது. எனவே, தேவைக்கேற்றார் போல், உரிய காண்பு நிலை பாங்கினை பயன்படுத்த வேண்டும்.

private என வகைப்படுத்தப்பட்ட அடிப்படை இனக்குழுவின் பண்புக்கூறுகளை தருவிக்கப்பட்ட இனக்குழுவின் உறுப்புகள் மட்டுமே பயன்படுத்திக்கொள்ள முடியும்.

ஆனால் அடிப்படை இனக்குழுவின் பண்புக்கூறுகள் தருவிக்கப்பட்ட இனக்குழுவின் அடிப்படையில் உருவாக்கப்பட்ட இனக்குழுவிற்கு இது பொருந்தாது.

protected என வகைப்படடுத்தப்பட்ட தருவிக்கப்பட்ட இனக்குழுவின் பண்புக்கூறுகளை தருவிக்கப்பட்ட இனக்குழுவின் உறுப்புகள் மட்டுமே எடுத்தாள முடியும். இனக்குழுவிற்கு வெளியே அதனை எடுத்தாள முடியாது.

public என வகைப்படுத்தப்பட்ட அடிப்படை இனக்குழுவின் பண்புக்கூறுகளை தருவிக்கப்பட்ட இனக்குழு மட்டுமல்லாது, இனக்குழுவுக்கு வெளியேயும் எடுத்தாள முடியும்.

16.6 மரபுரிமம், ஆக்கிகள் மற்றும் அழிப்பிகள்

ஒரு தருவிக்கப்பட்ட இனக்குழுவின் பொருளை உருவாக்கும் போது, நிரல் பெயர்ப்பி முதலில் அடிப்படை இனக்குழுவின் ஆக்கியை அழைக்கும், அதன்பின் தருவிக்கப்பட்ட இனக்குழுவின் ஆக்கியை அழைக்கும். ஏனெனில், தருவிக்கப்பட்ட இனக்குழு உறுப்புகள் அடிப்படை இனக்குழுவின் மீது கட்டமைக்கப்பட்டுள்ளது. தருவிக்கப்பட்ட இனக்குழுவின் பொருளின் பயன்பாடு முடிவுக்கு வரும்போது தருவிக்கப்பட்ட இனக்குழுவின் அழிப்பி முதலில் செயல்படுத்தப்பட்டு, அடுத்து அடிப்படை இனக்குழுவின் அழிப்பி இயக்கப்படும்.

எடுத்துக்காட்டு நிரல் 16.7 ஆக்கிகள், அழிப்பிகள் வரிசைமுறையை விளக்கிக்காட்டுகிறது.

```
#include<iostream>
using namespace std;
class base
{
public:
base()
{
cout<<"\nConstructor of base class...";
}
~base()
{
cout<<"\nDestructor of base class.... ";
}
};
class derived:public base
{
public :
derived()
{
cout << "\nConstructor of derived ...";
}
~derived()
{
cout << "\nDestructor of derived ...";
}
};
class derived1 :public derived
{
public :
derived1()
{
cout << "\nConstructor of derived1 ...";
}
~derived1()
{
cout << "\nDestructor of derived2 ...";
}
};
```


Chapter 16.indd 280 03/09/18 5:51 PM

```
int main()
{
derived1 x;
return 0;
}
```

வெளியீடு:

```
Constructor of base class...
Constructor of derived ...
Constructor of derived1 ...
Destructor of derived2 ...
Destructor of derived ...
Destructor of base class....
```

குறிப்பு

ஆக்கிகள் மரபுரிமம் பெற்ற இனக்குழுக்களின் வரிசையில் இயக்கப்படுகின்றன. அதாவது, அடிப்படை இனக்குழுவில் தொடங்கி இறுதியாக தருவிக்கப்பட்ட இனக்குழு வரையில் அழிப்பிகள் தலைகீழ் வரிசையில் இயக்கப்படும்.

மரபுரிமத்தில் ஆக்கிகள் செயல்படுத்தப்படும் முறை பற்றிய சில குறிப்புகள்

- அடிப்படை இனக்குழுவின் ஆக்கிகள் முதலில் இயக்கப்பட்டு, அடுத்து தருவிக்கப்பட்ட இனக்குழுவின் ஆக்கிகள் இயக்கப்படும்.
- தருவிக்கப்பட்ட இனக்குழு அடிப்படை இனக்குழுவின் ஆக்கியை மரபுவழி பெற முடியாது. ஆனால் அடிப்படை இனக்குழுவின் ஆக்கியை கீழ்கண்ட தருவிக்கப்பட்ட இனக்குழு வரையறுப்பின் மூலம் அழைக்கலாம்.
- அடிப்படை இனக்குழு :: அடிப்படை_இனக்குழு _ஆக்கி().
- பல அடிப்படை இனக்குழுக்கள் இருப்பின், அதன் இயக்கம் இடது பக்க வரிசை முறையில் தொடங்கப்படும்.
- பலநிலை மரபுரிமத்தில், ஆக்கிகள் மரபுரிமம் பெற்ற வரிசைமுறையில் இயக்கப்படும்.

குறிப்பு

தருவிக்கப்பட்ட இனக்குழுப் பொருளின் அளவு = அனைத்து அடிப்படை இனக்குழுவின் தரவு உறுப்புகளின் அளவு + தருவிக்கப்பட்ட இனக்குழுவில் உள்ள அனைத்துத் தரவு உறுப்புகளின் அளவு.

Chapter 16.indd 281 03/09/18 5:51 PM

16.7 மரபுரிமம் மற்றும் அணுகியல்பு கட்டுப்பாடு

ஒரு தருவிக்கப்பட்ட இனக்குழுவை அறிவிக்கும் போது, தருவிக்கப்பட்ட இனக்குழுவின் அடிப்படை பட்டியலில் ஒவ்வொரு அடிப்படை இனக்குழுவின் முன்னொட்டாக காண்பு நிலைப் பாங்கு இடம் பெற வேண்டும். இது அடிப்படை இனக்குழுவின் தனிப்பட்ட உறுப்புகளை அணுகும் பண்பினை மாற்றி அமைக்காது ஆனால் தருவிக்கப்பட்ட இனக்குழுவின் உறுப்புகள் அடிப்படை இனக்குழுவின் உறுப்புகளை சில கட்டுப்பாடுடன் அனுமதிக்கிறது.

முன்னரே அறிந்தபடி மூன்று காண்பு நிலை பாங்குகளில், ஏதேனும் ஒன்றைப் பயன்படுத்தி இனக்குழுக்களை தருவித்துக் கொள்ளலாம்.

- public என்ற காண்புநிலை கொண்ட அடிப்படை இனக்குழுவில், public மற்றும் protected என வகைப்படுத்தப்பட்ட உறுப்புகள், தருவிக்கப்பட்ட இனக்குழுவிலும் public மற்றும் protected உறுப்புகளாகவே இருக்கும்.

- protected என்ற காண்பு நிலை கொண்ட அடிப்படை இனக்குழுவில், public மற்றும் protected என வகைப்படுத்தப்பட்ட உறுப்புகள், தருவிக்கப்பட்ட இனக்குழுவில் protected உறுப்புகளாக இருக்கும்.

- private என்ற காண்பு நிலை கொண்ட அடிப்படை இனக்குழுவில், public மற்றும் protected என வகைப்படுத்தப்பட்ட உறுப்புகள் தருவிக்கப்பட்ட இனக்குழுவில் private உறுப்புகளாக ஆகிறது.

இவை அனைத்திலும் அடிப்படை இனக்குழுவில் private என வகைப்படுத்தப்பட்ட உறுப்புகள், private உறுப்புகளாகவே கருதப்படுவதால் தருவிக்கப்பட்ட இனக்குழுவில் அவை பயன்படுத்த முடியாது. எனினும் அடிப்படை இனக்குழுவின் public மற்றும் protected உறுப்பு செயற்கூறினைப் பயன்படுத்தி தருவிக்கப்பட்ட இனக்குழு இவற்றை மறைமுகமாக அணுகலாம். ஏனெனில் அடிப்படை இனக்குழுவில் இடம் பெறும் private உறுப்புகளை அணுக இவற்றிற்கு சிறப்பு உரிமை உண்டு

எடுத்துக்காட்டு 16.8 அடிப்படை இனக்குழுவின் உறுப்பினை தருவிக்கப்பட்ட இனக்குழு அணுகும் முறை

```
#include<iostream>
using namespace std;
class add
{
int num1,sum; // private data members can not be derived by derived class.
protected:

 int num2;
public:
add()
{
 num1= num2= sum=0;
cout<<"\n Add constructor .. ";
}
```

```
accept()
{
cout<<"\n Enter two numbers .. ";
cin>>num1>>num2;
}
plus()
{
sum = num1 + num2;
cout<<"\n The sum of two numbers is .. "<< sum;
}
int difference(){ return num1-num2;}        //return an integer value
~add()
 {cout<<endl<<"Add destructor"; }
};
class subtract :public add                         // subtract  derived from add
{
int sub;
public:
subtract()
{
sub = 0;
cout<<"\n Subtract constructor .. ";
}
minus()
{
  accept(); // member function of base class can access is data members

sub= difference();

cout<<"\n The difference of two numbers are ... "<< sub;
}
~subtract()
 {cout<<endl<<"Subtract destructor"; }
};
int main()
{
subtract s;
int choice = 0;
cout<<"\n Enter your choice ";
cout<<" \n1. Add..\n2. Subtract ..";
cin>>choice;
```


Chapter 16.indd   283   03/09/18   5:51 PM

```
switch( choice )
{
case 1:
s.accept();
s.plus();
break;
case 2:
s.minus();
break;
}
return 0;
}
```

ஓர் உறுப்பு செயற்கூறு மற்றொரு உறுப்பு செயற்கூறினை புள்ளி செயற்குறி மற்றும் பொருள் இல்லாமல் அழைக்க முடியும்.

வெளியீடு:

```
    Add constructor ..
    Subtract constructor ..
    Enter your choice
        1. Add..
        2. Subtract .. 2
    Enter two numbers ..  20 10
    The difference of two numbers are ... 10
    Subtract destructor
    Add destructor
```

மேற்கண்ட நிரலில், **num1** என்ற தரவு உறுப்புகள் **private add** இனக்குழுவில் காண்பு நிலை கொண்டுள்ளதால், **subtract** இனக்குழு மரபுவழி பெற முடியாது. ஆனால், **add** இனக்குழுவின் **public** மற்றும் **protected** உறுப்புகளை **subtract** இனக்குழு மரபுவழி பெற முடியும். ஆகவே மரபு வழி பெறப்பட்ட **accept** () செயற்கூறினைப் பயன்படுத்தி **num1** என்ற மாறியின் மதிப்பு பெற்றுக் கொள்ளப்படுகிறது. மேலும் மரபுவழி பெறப்பெற்ற **difference**() செயற்கூறினைப் பயன்படுத்தி மதிப்புகள் கழிக்கப்பட்டு விடையை **subtract** இனக்குழுவிற்கு அனுப்பி வைக்கப்படுகிறது.

16.7.1 Public காண்புநிலையுடன் தருவிக்கப்பட்ட இனக்குழுவின் அணுகியல்பு கட்டுப்பாடு

public காண்புநிலையுடன் தருவிக்கப்பட்ட இனக்குழுவில் Public மற்றும் protected உறுப்புகள் என அடிப்படை இனக்குழுவில் வரையறுப்பின் அவை தருவிக்கப்பட்ட இனக்குழுவிலும் Public மற்றும் protected உறுப்புகளாகவே நிலவும். தருவிக்கப்பட்ட இனக்குழுவின் பொருள் அதன் Public உறுப்புகளை அணுகுவதைப் போன்றே அடிப்படை இனக்குழுவின் Public உறுப்புகளை அணுகும்.

எடுத்துக்காட்டு நிரல் 16.9 Public காண்புநிலையுடன் தருவிக்கப்பட்ட இனக்குழுவை விளக்குகிறது.

```
#include<iostream>
#include<string>
using namespace std;
class Employee
{
   private:
         char name[50];
         int code;
   public:
         void getinfo();
         void dispinfo();
};
class staff: public Employee
{
         private:
         int ex;
         public:
         void getdata();
         void display();
};
void Employee::getinfo()
{
         cout<<"Name:";
         gets(name);
         cout<<"Code:";
         cin>>code;
}
void Employee::dispinfo()
{
         cout<<"Name:"<<name<<endl;
         cout<<"Code:"<<code<<endl;
}
void staff::getdata()
{
         cout<<"Experience:";
         cin>>ex;
}
```


Chapter 16.indd 285 03/09/18 5:51 PM

```
void staff::display()
{
   cout<<"Experience:"<<ex<<" Years"<<endl;
}
nt main()
{
         staff s;
         cout<<"Enter data"<<endl;
         s.getinfo();             //Derived member function
         s.getdata();             //Defined member function
         cout<<endl<<endl<<"\tDisplay Data"<<endl;
         s.dispinfo();            //Derived member function
         s.display();             //Defined member function
return 0;
}v
```

வெளியீடு:

```
         Enter data
         Name:USHA
         Code:1201
         Experience: 30
         Display Data
         Name:USHA
         Code:1201
         Experience:30 Years
```

மேற்கண்ட நிரலில் இனக்குழுவின் பொருளானது தருவிக்கப்பட்ட செயற்கூறினை அணுகமுடியும். ஏனெனில் “staff” என்ற இனக்குழுவானது public காண்புநிலையுடன் தருவிக்கப்பட்டுள்ளது.

குறிப்பு

> Public காண்புநிலையுடன் மரபுவழியில் இனக்குழு பெறப்பட்டால், அடிப்படை இனக்குழுவின் குறிப்பிட்ட உறுப்புகளை அணுகுவதை மறுக்க முடியாது.

16.7.2 private காண்புநிலையுடன் தருவிக்கப்பட்ட இனக்குழுவின் அணுகியல்பு கட்டுப்பாடு

Private காண்புநிலையுடன் தருவிக்கப்பட்ட இனக்குழுவில், அடிப்படை இனக்குழுவில் உள்ள public மற்றும் protected உறுப்புகள், தருவிக்கப்பட்ட இனக்குழுவில் private உறுப்புகளாக மாறிவிடுகின்றன. எனவே தருவிக்கப்பட்ட இனக்குழுவின் பொருளை கொண்டு தருவிக்கப்பட்ட உறுப்புகளை அணுக முடியாது. public காண்புநிலையுடன் வரையறுக்கப்பட்ட உறுப்புகளை கொண்டு தருவிக்கப்பட்ட உறுப்புகளை அழைக்கலாம்.

Chapter 16.indd 286 03/09/18 5:51 PM

எடுத்துக்காட்டு 16.10 காண்புநிலையுடன் பெறப்பட்ட தருவிக்கப்பட்ட இனக்குழு

```
#include<iostream>
#include<string>
using namespace std;
class Employee
{
   private:
           char name[50];
           int code;
   public:
           void getinfo();
           void dispinfo();
};
class staff: private Employee
{
   private:
           int ex;
   public:
           void getdata();
           void display();
};
void Employee::getinfo()
{
           cout<<"Name:";
           gets(name);
           cout<<"Code:";
           cin>>code;
}
void Employee::dispinfo()
{
           cout<<"Name:"<<name<<endl;
           cout<<"Code:"<<code<<endl;
}
void staff::getdata()
{
    getinfo();              //invoked inside
cout<<"Experience:";
   cin>>ex;
}
void staff::display()
{
  dispinfo();      //member function called inside another member function
```


Chapter 16.indd 287 03/09/18 5:51 PM

```
cout<<"Experience:"<<ex<<" Years"<<endl;
}
int main()
{
   staff s;
   cout<<"Enter data"<<endl;
         s.getdata();
   cout<<endl<<endl<<"\tDisplay Data"<<endl;
         s.display();
   return 0;
}
```

வெளியீடு:

```
Enter data
Name: BALAMURUGAN
Code: 1201
Experience: 30
Display Data
Name:  BALAMURUGAN
Code: 1201
Experience: 30 Years
```

public காண்புநிலை மற்றும் private காண்புநிலை எடுத்துக்காட்டு நிரல்களில் இடம் பெற்ற இரண்டும் ஒன்று போல் இருந்தாலும் அவற்றை வரையறுக்கும் முறை மட்டும் இருக்கிறது. இது எதனால் என்றால், தருவிக்கப்பட்ட இனக்குழுவின் private உறுப்புகளை அந்த இனக்குழுவின் பொருளால் அணுக முடியாது. மேற்கண்ட நிரலில் “staff” என்ற இனக்குழு private() காண்புநிலையுடன் தருவிக்கப்பட்டதால், getinfo() மற்றும் dispinfo() செயற்கூறுகள் “staff” இனக்குழுக்கு Private யாக கருதப்படுகின்றன. இவ்விரு உறுப்புச் செயற்கூறுகளை அணுக அவை public உறுப்பு செயற்கூறான getdata() மற்றும் display() செயற்கூறினுள்ளே அழைக்கப்பட்டுள்ளன.

குறிப்பு

உறுப்பு செயற்கூறுகள், private என்று வகைப்படுத்தப்பட்ட உறுப்புகளை அணுகலாம்.

16.8 அடிப்படை இனக்குழுவின் செயற்கூறுகளை தருவிக்கப்பட்ட இனக்குழுவில் மேனிடல் / நிழலிடல் (Overriding / Shadowing)

மரபுரிமத்தில், சில சமயங்களில் அடிப்படை இனக்குழுக்களும், தருவிக்கப்பட்ட இனக்குழுக்களும் ஒரே பெயரில் உறுப்பு செயற்கூறினை பெற்றிருக்க முடியும். தருவிக்கப்பட்ட இனக்குழுவின் பொருள் பணி மிகுக்கப்பட்ட உறுப்பு செயற்கூறினை அழைக்கும் போது எந்த செயற்கூறினை செயல்படுத்துவது என்ற குழப்பத்தினை நிரல்பெயர்ப்பிக்கு ஏற்படுத்துகிறது. தருவிக்கப்பட்ட இனக்குழுவின் உறுப்பு செயற்கூறினுக்கு அடிப்படை இனக்குழு உறுப்பு செயற்கூறினை காட்டிலும் அதிக முன்னுரிமை உள்ளது. அடிப்படை இனக்குழுவின் உறுப்பு செயற்கூறுகள் தருவிக்கப்பட்ட இனக்குழுவின் உறுப்புச் செயற்கூறுகளின் பெயரை

கொண்டிருந்தால், அது அடிப்படை இனக்குழுவின் உறுப்பு செயற்கூறுகளின் நிழலாக கருதப்படுகிறது. வரையெல்லை தீர்மானிப்பு செய்குறி (::) கொண்டு இந்த சிக்கலை தீர்க்கலாம்.

எடுத்துக்காட்டு நிரல் 16.11 தருவிக்கப்பட்ட இனக்குழுவில் வரையெல்லை தீர்மானிப்பு செயற்கூறின் பயன்பாட்டை விளக்குகிறது.

```
#include<iostream>
#include<string>
using namespace std;
class Employee
{
  private:
    char name[50];
    int code;
  public:
    void getdata();
    void display();
};
class staff: public Employee
{
  private:
    int ex;
  public:
       void getdata();
    void display();
};
void Employee::display()
{
       cout<<"Name:"<<name<<endl;
       cout<<"Code:"<<code<<endl;
}
void staff::display()
{
       Employee :: display();//overriding
       cout<<"Experience:"<<ex<<" Years"<<endl;
}
```


Chapter 16.indd 289 03/09/18 5:51 PM

```
int main()
{
        staff s;
        cout<<"Enter data"<<endl;
        s.getdata();
        cout<<endl<<endl<<"\tDisplay Data"<<endl;
        s.display();
        return 0;
}
```

வெளியீடு

```
        Enter data
        Name: SUGANYA
        Code: 1201
        Experience: 30
        Display Data
        Name: SUGANYA
        Code:1201
        Experience:30 Years
```

மேற்கண்ட நிரலில் getdata() மற்றும் display() என்ற இரண்டு செயற்கூறுகளும் அடிப்படை இனக்குழு மற்றும் தருவிக்கப்பட்ட இனக்குழுவில் வரையறுக்கப்பட்டிருக்கிறது. எனவே, தருவிக்கப்பட்ட இனக்குழு "staff", "employee" இனக்குழுவின் பண்புக்கூறுகளை மரபுவழி பெறும் போது, அது இரண்டு getdata() மற்றும் display() கொண்டிருக்கும். வரையெல்லை தீர்மானிப்பு செயற்குறியை (::) கொண்டு தருவிக்கப்பட்ட getdata(), display() செயற்கூறினையும், வரையறுக்கப்பட்ட getdata(), display() என்ற செயற்கூறினையும் வேறுபடுத்திக்காட்டலாம். இந்த செயற்குறியை அடிப்படை இனக்குழுவின் பெயருக்கும் உறுப்பு செயற்கூறின் பெயருக்கும் நடுவில் கொடுக்கப்பட வேண்டும்.

குறிப்பு

> தருவிக்கப்பட்ட இனக்குழுவின் உறுப்பு செயற்கூறும், அடிப்படை இனக்குழுவின் உறுப்பு செயற்கூறும் ஒரே பெயரை பெற்றிருந்தால், தருவிக்கப்பட்ட இனக்குழுவின் உறுப்பு செயற்கூறுகள் அடிப்படை இனக்குழுவின் மரபுவழி பெறப்பட்ட செயற்கூறுகளை நிழலிடும்/மறைக்கும். இதை செயற்கூறு மேலிடல் என்கிறோம். இந்த சிக்கலை தீர்க்க அடிப்படை இனக்குழு பெயர் அடுத்து :: மற்றும் உறுப்பு செயற்கூறு பெயர் குறிப்பிட வேண்டும்.

16.8.1 this சுட்டு

'this' சுட்டு ஓர் மாறிலி சுட்டு ஆகும்; அது நடப்பு பொருளின் நினைவக முகவரியை நிறுத்திவைக்கிறது. இது நடப்பில் அழைக்கப்படும் பொருளை அடையாளம் கண்டுகொள்கிறது. இது, உறுப்புச் செயற்கூறுகளின்

Chapter 16.indd 290 03/09/18 5:51 PM

அளபுரு மாறியின் பெயரும், உறுப்பு தரவுகளின் பெயரும் ஒன்றாக இருக்கும் போது பயனுள்ளதாக இருக்கும். தரவு உறுப்புகளை அடையாளம் காண "this→ தரவு உறுப்பின் பெயர்" என்று கொடுக்கப்பட வேண்டும். பின்வரும் நிரல் 16.12 'this' சுட்டின் பயன்பாட்டை எடுத்துக்காட்டுகிறது.

எடுத்துக்காட்டு நிரல் 16.12 this சுட்டு பயன்பாடு

```
#include<iostream>
using namespace std;
class T
{
  public:
int x;
    void foo()
  {
     x = 6;      // same as this->x = 6;
     this->x = 5; // explicit use of this->
     cout<<endl<<x<<" "<<this->x;
  }
  void foo(int x)  // parameter x shadows the member with the same name
  {
      this->x = x;              // unqualified x refers to the parameter.'this->' required
for disambiguation
     cout<<endl<<x<<" "<<this->x;
  }};
int main()
  {
        T t1,t2;
        t1.foo();
        t2.foo();
        }
```

வெளியீடு

```
        5  5
        5  5
        --------------------------------
        Process exited after 0.1 seconds with return value 0
        Press any key to continue . . .
```

எடுத்துக்காட்டு நிரல் 16.13 this சுட்டு - மற்றுமொரு எடுத்துக்காட்டு

```
#include <iostream>
using namespace std;

class Container {
  public:
    // Constructor definition
    Container(double l = 2.0, double b = 2.0, double h = 2.0) {
cout<<"Constructor called." << endl;
      length = l;
      breadth = b;
      height = h;
    }
    double Volume() {
      return length * breadth * height;
    }
int compare(Container container) {
      return this->Volume() >Container.Volume();
    }
  private:
    double length;     // Length of a Container
    double breadth;    // Breadth of a Container
    double height;     // Height of a Container
};
int main(void) {
  Container Container1(3.3, 1.2, 1.5);    // Declare Container1
  Container Container2(8.5, 6.0, 2.0);    // Declare Container2
  if(Container1.compare(Container2)) {
    cout << "Container2 is smaller than Container1" <<endl;
  } else {
    cout << "Container2 is equal to or larger than Container1" <<endl;
  }
    return 0;
}
```

வெளியீடு

```
Constructor called.
Constructor called.
Container2 is equal to or larger than Container1
--------------------------------
Process exited after 0.09358 seconds with return value 0
Press any key to continue . . .
```


Chapter 16.indd 292 03/09/18 5:51 PM

நினைவில் கொள்க

- ஏற்கெனவே இருக்கும் ஓர் இனக்குழுவின் அடிப்படையில் புதிய இனக்குழுவை தருவித்தல் மரபுரிமம் எனப்படும்.
- குறிமுறையின் மறுபயனாக்கம் மரபுரிமத்தின் முக்கிய அனுகூலமாகும்.
- தருவிக்கப்பட்ட இனக்குழு அடிப்படை இனக்குழுவின் பண்புக்கூறுகளை மரபுரிமையாக பெறுகிறது. தருவிக்கப்பட்ட இனக்குழுக்கள் சக்தி மிக்கவை. தருவிக்கப்பட்ட இனக்குழுவில் கூடுதலான பண்புக் கூறுகளையும், வழிமுறைகளையும் சேர்த்து அதன் செயலாற்றலை அதிகரிக்க முடியும்.
- மரபுரிமம் ஒருவழி மரபுரிமம், பல வழி மரபுரிமம், பலநிலை மரபுரிமம், படிநிலை மரபுரிமம், கலப்பு மரபுரிமம் என பல வகைப்படும்.
- ஒரு அடிப்படை இனக்குழுவிலிருந்து மட்டும் தருவிக்கப்படும் இனக்குழு ஒருவழி மரபுரிமம் எனப்படும்.
- ஓர் இனக்குழு பல அடிப்படை இனக்குழுவிலிருந்து தருவிக்கப்படுமாயின் அது பலவழி மரபுரிமம் எனப்படும்.
- தருவிக்கப்பட்ட ஓர் இனக்குழுவை அடிப்படையாகக் கொண்டு இன்னொரு தருவிக்கப்பட்ட இனக்குழுவை உருவாக்குவது பலநிலை மரபுரிமம் ஆகும். மரபுரிமத்தின் மாறக்கூடிய தன்மை இதில் பிரதிபலிக்கிறது.
- ஒன்றுக்கு மேற்பட்ட இனக்குழுக்கள் ஒரே ஒரு அடிப்படை இனக்குழுவிலிருந்து தருவிக்கப்பட்டால், அது படிமுறை மரபுரிமம் எனப்படுகிறது.
- ஒன்றுக்கு மேற்பட்ட மரபுரிம வகைகளைக் கொண்டு உருவாக்கப்பட்ட மரபுரிமம் கலப்பு மரபுரிமம் எனப்படும்.
- பலவழி மரபுரிமத்தில், தருவிக்கப்பட்ட இனக்குழுவில் அடிப்படை இனக்குழுக்கள் அறிவிக்கப்பட்ட வரிசையில் கட்டமைக்கப்படுகின்றன.
- ஒரு துணை இனக்குழு public, private அல்லது protected என்ற காண்பு நிலையுடன் தருவித்துக் கொள்ளலாம்.
- private என வகைப்படுத்தப்பட்ட இனக்குழுவின் உறுப்புகள் மரபு வழி பெற முடியாது.
- ஓர் அடிப்படை இனக்குழு public என்னும் அணுகியல்புடன் தருவிக்கப்படும்போது, அடிப்படை இனக்குழுவின் public உறுப்புகள் public உறுப்புகளாகவும், protected உறுப்புகள் protected உறுப்புகளாகவும் தருவிக்கப்பட்ட இனக்குழுவில் இருக்கும்.
- ஓர் இனக்குழு private அணுகியல்புடன் தருவிக்கப்படும்போது, அடிப்படை இனக்குழுவின் public மற்றும் protected உறுப்புகள், தருவிக்கப்பட்ட இனக்குழுவில் private உறுப்புகளாக கருதப்படும்.
- Protected காண்புநிலையில் ஓர் இனக்குழு தருவிக்கப்படும்போது, அடிப்படை இனக்குழுவின் public மற்றும் protected உறுப்புகள் தருவிக்கப்பட்ட இனக்குழுவில் protected உறுப்புகளாக கருதப்படும்.
- அடிப்படை இனக்குழவின் ஆக்கிகள் மற்றும் அழிப்பிகள் மரபுவழியாக பெறப்படுவதில்லை. ஆனால் தருவிக்கப்பட்ட இனக்குழவில் ஒரு பொருளை உருவாக்கும்போது, அடிப்படை இனக்குழுவின் ஆக்கிகள் தாமாகவே அழைக்கப்பட்டுவிடும்.
- அழிப்பிகள் முன்பின் வரிசையில் அழைக்கப்படும். தருவிக்கப்பட்ட இனக்குழுவின் அழிப்பிகள் முதலில் இயக்கப்பட்டு, பின் அடிப்படை இனக்குழுவின் அழிப்பிகள் அழைக்கப்படும்.
- தருவிக்கப்பட்ட இனக்குழு பொருளின் அளவு=அனைத்து அடிப்படை இனக்குழுவின் தரவு உறுப்புகளின் அளவு + அனைத்து தருவிக்கப்பட்ட இனக்குழுவின் தரவு உறுப்புகளின்அளவு
- வரையெல்லை தீர்மானிப்பு செயற்குறி (::) உறுப்புகளின் மேலிடலை (overriding) தவிர்க்கும்.
- 'this' சுட்டு நடப்பு பொருளின் உறுப்புகளை குறிக்க பயன்படுகிறது.

Chapter 16.indd 293 03/09/18 5:51 PM

மதிப்பீடு

N2H6U8

பகுதி அ

I சரியான விடை யை தேர்வு செய்யவும் :

1. பின்வருவனவற்றுள் எது ஏற்கெனவே உள்ள இனக்குழுவின் அடிப்படையில் புதிய இனக்குழுவை தருவிக்கும் முறையாகும் ?

   அ) பல்லுருவாக்கம் ஆ) மரபுரிமம்

   இ) உறை பொதியாக்கம் ஈ) மீ – இனக்குழு

2. பின்வருவனவற்றுள் எது school' என்ற அடிப்படை இனக்குழுவிலிருந்து 'student' என்ற இனக்குழுவை தருவிக்கும் ?

   அ) school : student ஆ) class student : public school

   இ) student : public school ஈ) class school : public student

3. மாறக் கூடிய தன்மையை பிரதிபலிக்கும் மரபுரிம வகை

   அ) ஒருவழி மரபுரிமம் ஆ) பலவழி மரபுரிமம்

   இ) பலநிலை மரபுரிமம் ஈ) கலப்பு மரபுரிமம்

4. அடிப்படை இனக்குழுவின் பண்புகளை தருவிக்கப்பட்ட இனக்குழுவில் மட்டும் கிடைக்கப் பெற்று, ஆனால் தருவிக்கப்பட்ட இனக்குழுவை அடிப்படையாகக் கொண்டு தருவிக்கப்படும் இனக்குழுவில் கிடைக்கப்படாமல் இருக்க எந்த காண்புநிலை பாங்கினைப் பயன்படுத்த வேண்டும்?

   அ) private ஆ) public இ) protected ஈ) இவையனைத்தும்

5. மரபுரிமம் செயல்முறையில் புதிய இனக்குழு எதிலிருந்து உருவாக்கப்படுகிறது ?

   அ) அடிப்படை இனக்குழு ஆ) அருவமாக்கம்

   இ) தருவிக்கப்பட்ட இனக்குழு ஈ) செயற்கூறு

6. தருவிக்கப்பட்ட ஓர் இனக்குழுவை அடிப்படையாக கொண்டு இன்னொரு தருவிக்கப்பட்ட இனக்குழுவை உருவாக்குவது

   அ) பலவழி மரபுரிமம் ஆ) பலநிலை மரபுரிமம்

   இ) ஒருவழி மரபுரிமம் ஈ) இரட்டை மரபுரிமம்

7. பின்வருவனவற்றுள் எது மரபுரிமம் பெற்ற வரிசையில் இயக்கப்படுகிறது?

   அ) அழிப்பி ஆ) உறுப்பு செயற்கூறு இ) ஆக்கி ஈ) பொருள்

Chapter 16.indd 294 03/09/18 5:51 PM

8. பின்வருவனவற்றுள் எது மரபுரிமம் சார்ந்த சரியான கூற்று ?

அ) private அணுகியல்பு கொண்ட தருவிக்கப்பட்ட இனக்குழு அடிப்படை இனக்குழுவின் private உறுப்புகளை மரபுவழி பெறுகிறது.

ஆ) private அணுகியல்பு கொண்ட தருவிக்கப்பட்ட இனக்குழு அடிப்படை இனக்குழுவின் private உறுப்புகளை மரபுவழி பெறாது.

இ) அடிப்படை இனக்குழுவின் public உறுப்புகள், தருவிக்கப்பட்ட இனக்குழுவில் மரபுவழி பெறப்படும். ஆனால் அணுக முடியாது.

ஈ) அடிப்படை இனக்குழுவின் protected உறுப்புகள், இனக்குழுவிற்கு வெளியே மரபுவழி பெறப்படும். ஆனால் அணுக முடியாது.

9. பின்வரும் இனக்குழு அறிவிப்பின் அடிப்படையில், கீழ்காணும் வினாக்களுக்கு விடையளி. (9.1 லிருந்து 9.5 வரை)

```
class vehicle
{ int wheels;
public:
void input_data(float,float);
void output_data();
protected:
int passenger;
 };
class heavy_vehicle : protected vehicle {
int diesel_petrol;
protected:
int load;
protected:
int load;
public:
voidread_data(float,float)
voidwrite_data(); };
class bus: private heavy_vehicle {
charTicket[20];
public:
voidfetch_data(char);
voiddisplay_data(); };
};
```

9.1 heavy vehicle என்னும் இனக்குழுவின் அடிப்படை இனக்குழுவை குறிப்பிடுக.

அ) Bus ஆ) heavy-vehicle இ) vehicle ஈ) (அ) மற்றும் (இ)

9.2 display data ( ) என்னும் செயற்கூறு மூலம் அணுக முடிகிற தரவு உறுப்புகளை குறிப்பிடுக

அ) passenger ஆ) load இ) Ticket ஈ) all of these

9.3 bus இனக்குழுவின் பொருள், அணுக கூடிய தரவு உறுப்பு செயற்கூறுகளை குறிப்பிடுக.

அ) input_data() ஆ) read_data(),output data()write_data()

இ) fetch_data() ஈ) all of these display_data()

9.4 Bus இனக்குழுவில் public காண்புநிலையுடன் வரையறுக்கப்பட்ட உறுப்பு செயற்கூறு யாது

அ) input_data() ஆ) read_data(),output data()write_data()

இ) fetch_data() ஈ) all of these display_data()

9.5 heavy-vehicle இனக்குழுவின் பொருள்களால் அணுகக்கூடிய உறுப்பு செயற்கூறு யாது?

அ) void input data (int, int) ஆ) void output data ( )

இ) void read data (int, int) ஈ) both (அ) & (ஆ)

10)

```
class x
{  int a;
public :
x()

  {}
};
class y
{  x x1;
public :
y(){}
};
class z : public y,x
{
int b;
public:
z(){}
}z1;
```

Z1 என்ற பொருள் ஆக்கிகளை எந்த வரிசை முறையில் அழைக்கும் ?

(A) z , y,x,x (B) x,y,z,x (c) y,x,x,z (D)x,y,z

பகுதி– ஆ

II குறுவினாக்கள் (2 மதிப்பெண்கள்)

1. மரபுரிமம் என்றால் என்ன ?

2. அடிப்படை இனக்குழு என்றால் என்ன ?

3. தருவிக்கப்பட்ட இனக்குழு ஏன் சக்தி வாய்ந்த இனக்குழு என்று கருதப்படுகிறது ?

4. பல அடிப்படை இனக்குழுக்கள் கொண்ட பலநிலை மற்றும் பலவழி மரபுரிமம் எந்த வகையில் வேறுபடுகிறது?

5. **public** மற்றும் **private** காண்பு நிலை பாங்கு வேறுபாடு தருக.

பகுதி– இ

III சிறுவினாக்கள் (3 மதிப்பெண்கள்)

1. ஓர் இனக்குழுவை தருவிக்கும்போது, கவனத்தில் கொள்ள வேண்டியவை யாவை?

2. **private** காண்புநிலையில் இருக்கும் உறுப்புகளுக்கும், **public** காண்புநிலையில் இருக்கும் உறுப்புகளுக்கும் உள்ள வேறுபாடுகள் யாவை ?

3. நிரல் முறையின் மறுபயனாக்கத்திற்கு உதவுகின்ற பல்லுருவாக்கத்திற்கும் மரபுரிமத்திற்கும் உள்ள வேறுபாடுகள் யாவை?

4. மேலிடல் என்றால் என்ன?

5. மரபுரிமத்தில் இயக்கப்படும் ஆக்கிகள் மற்றும் அழிப்பிகள் பற்றி சிறுகுறிப்பு வரைக.

பகுதி – ஈ

IV பெருவினாக்கள் (5 மதிப்பெண்கள்)

1. மரபுரிமத்தின் பல்வேறு வகைகளை விவரி.

2. பல்வேறு காண்புநிலை பாங்கினை வரைபடத்தை கொண்டு விளக்குக.

Chapter 16.indd 297 03/09/18 5:51 PM

3.

```
#include<iostream>
#include<string.h>
#include<stdio.h>
using  name spacestd;
class publisher
{
char pname[15];
char hoffice[15];
char address[25];
double turnover;
protected:
char phone[3][1O];
void register();
public:
publisher();
~publisher();
void enter data();
void disp data();
};
class branch
{
charbcity[15];
char baddress[25];
protected:
intno_of_emp;
public:
charbphone[2][10];
branch();
~branch();
void have data();
void give data();
};
class author: public branch, publisher
{
intaut_code;
charaname[2O];
float income;
public:
author();
~author();
voidgetdata();
voidputdata();
};
```


Chapter 16.indd 298 03/09/18 5:51 PM

கீழ்காணும் வினாக்களுக்கு மேற்கண்ட நிரலின் அடிப்படையில் விடையளிக்கவும்.

3.1 நிரல் குறிமுறையில் எந்தவகை மரபுரிமம் குறிப்பிடப்பட்டுள்ளது?

3.2 அடிப்படை இனக்குழுக்களின் காண்புநிலை பாங்கு யாது?

3.3 Author இனக்குழுவிற்கு பொருள்உருவாக்கப்படும்போது ஆக்கி, அழிப்பி இயக்கப்படும் வரிசைமுறையை எழுதுக.

3.4 அடிப்படை இனக்குழு(கள்) மற்றும் தருவிக்கப்பட்டஇனக்குழு(கள்) பெயரை குறிப்பிடுக.

3.5 பின்வரும் இனக்குழுவின் பொருள் எத்தனை பைட்டுகளை எடுத்துக்கொள்ளும்.

a. Publisher (b) branch (c) author

3.6 author இனக்குழுவின் பொருளாள் அணுகக்கூடிய தரவு உறுப்புகளின் பெயர்களை குறிப்பிடுக.

3.7 author இனக்குழுவின் பொருளால் அணுகக்கூடிய உறுப்பு செயற்கூறுகளின் பெயர்கள் குறிப்பிடுக.

3.8 author இனக்குழுவின் உறுப்பு செயற்கூறுகள் அணுகக்கூடிய தரவு உறுப்புகளின் பெயர்களை குறிப்பிடுக.

4. பின்வரும் **C++** நிரல் குறிமுறைக் கொண்டு, கீழ்காணும் வினாக்களுக்கு விடையளி

```
class Personal
{
int Class,Rno;
char Section;
protected:
char Name[20];
public:
personal();
void pentry();
voidPdisplay();
};
class Marks:private Personal
{
float M{5};
protected:
char Grade[5];
public:
Marks();
void M entry();
void M display();
};
class Result:public Marks
{
float Total,Agg;
public:
char FinalGrade, Commence[20];
Result();
void R calculate();
void R display();
}:
```


Chapter 16.indd 299 03/09/18 5:51 PM

4.1 நிரல் குறிமுறையில் எந்த வகை மரபுரிமம் குறிப்பிடப்பட்டுள்ளது?

4.2 அடிப்படை இனக்குழுக்களின் காண்புநிலை பாங்கினைகுறிப்பிடுக.

4.3 Result இனக்குழுவிற்கு பொருள் உருவாக்கப்படும்போது, ஆக்கி, அழிப்பி இயக்கப்படும் வரிசைமுறையை எழுதுக.

4.4 அடிப்படை இனக்குழு(கள்) மற்றும் தருவிக்கப்பட்டட இனக்குழு(கள்) பெயர்களை குறிப்பிடுக.

4.5 பின்வரும் இனக்குழுக்களின் பொருள் எத்தனை பைட்டுகள் எடுத்துக்கொள்ளும்?

(a) Personal (b) Marks (c) Result

4.6 Result இனக்குழுவின் பொருளால் அணுகக்கூடிய தரவு உறுப்புகளின் பெயர்களை குறிப்பிடுக.

4.7 Result இனக்குழுவின் பொருளால் அணுகக்கூடிய உறுப்பு செயற்கூறுகளின் பெயர்களை குறிப்பிடுக.

4.8 Result இனக்குழுவின் உறுப்பு செயற்கூறுகள் அணுகக்கூடிய தரவு உறுப்புகளின் பெயர்களை குறிப்பிடுக.

5. கீழ்காணும் நிரலுக்கு வெளியீட்டை எழுதுக.

```
#include<iostream>
using namespace std;
class A
{
protected:
int x;
public:
void show()
{
cout<<"x = "<<x<<endl;
}
A()
{
        cout<<endl<<" I am class A "<<endl;
}
~A()
{
        cout<<endl<<" Bye ";
}
};
class B : public A
{
{
protected:
int y;
public:
B(int x, int y)
{
```

```
this->x = x;            //this -> is used to denote the objects datamember
this->y = y;            //this -> is used to denote the objects datamember
}
B()
{
        cout<<endl<<" I am class B "<<endl;
}
~B()
{
        cout<<endl<<" Bye ";
}
void show()
{
cout<<"x = "<<x<<endl;
cout<<"y = "<<y<<endl;
}
};
int main()
{
AobjA;
B objB(30, 20);
objB.show();
return 0;
}
```

6. கீழ்கண்ட நிரலில் உள்ள பிழைகளை கண்டறிந்து பிழைதிருத்தம் செய்க,

வெளியீடு

```
-------------
15
14
13
```

நிரல் :

```
-------------
%include(iostream.h)
#include<conio.h>
Class A
{
public;
int a1,a2:a3;
Void getdata[]
```


Chapter 16.indd 301 03/09/18 5:51 PM

```
{
        a1=15;
        a2=13;a3=13;
}
}
Class B:: public A()
{
      PUBLIC
      voidfunc()
      {
        int b1:b2:b3;
        A::getdata[];
        b1=a1;
        b2=a2;
        a3=a3;
        cout<<b1<<'\t'<<b2<<'t\'<<b3;
      }
void main()
{
      clrscr()
      B der;
      der1:func();
      getch();
}
```

ஆய்வறிக்கை 1

இந்தியாவில் செயல்படுகின்ற அனைத்துவங்கிகளும் இந்திய ரிசர்வ் வங்கியினால் கட்டுப்படுத்தப் படுகின்றது. ரிசர்வ் வங்கி அனைத்து வங்கிகளும் பின்பற்ற வேண்டிய சில வழிகாட்டுதல்களை (எ.கா. குறைந்தபட்ச வட்டிவிகிதம், குறைந்தபட்ச அனுமதிக்கப்பட்ட இருப்புத்தொகை, அதிகபட்ச பணத்தை திரும்ப எடுக்கும் வரம்பு போன்றவை) வகுக்கிறது. எடுத்துக்காட்டாக, ரிசர்வ் வங்கி சேமிப்பு கணக்கிற்கு குறைந்தபட்ச வட்டி ஆண்டொன்றுக்கு 5% என நிர்ணயத்துள்ளது, எனினும் வங்கிகள்5% வட்டி விகிதம் அல்லது அதற்கு மேல் கொடுப்பதற்கு அனுமதியளிக்கப்படுகிறது.

மேற்கண்ட சூழலில் வங்கியின் செயல்பாட்டினை நடைமுறைப்படுத்துவதற்கு நிரல் ஒன்று எழுதுக.

குறிப்பு

- Customer, Account, RBI (அடிப்படை இனக்குழு) என்ற இனக்குழுக்களை உருவாக்கவும்.
- SBI,ICICI,PNB, IOB போன்று சில தருவிக்கப்பட்ட இனக்குழுக்களை உருவாக்கவும்.

ஒவ்வொரு இனக்குழுவிற்கும் தேவையான உறுப்பு மாறிகளையும், செயற்கூறுகளையும் விருப்பத்திற்கேற்ப உருவாக்கிக்கொள்ளவும்.

```
class Customer
{
//Personal Details ...
// Few functions ...
}
class Account
{
// Account Detail ...
// Few functions ...
}
}
class RBI
{
Customer c; //hasA relationship
Account a;      //hasA relationship
..
```

```
Public double GetInterestRate() {       }
Public double GetWithdrawalLimit() {          }
}
class SBI: public RBI
{
//Use RBI functionality or define own functionality.
}
class ICICI: public RBI
{
//Use RBI functionality or define own functionality.
}
class IOB: public RBI
{
//Use RBI functionality or define own functionality.
}
```

ஆய்வறிக்கை -2

1. Stock என்ற இனக்குழுவை வரையறுக்கவும். -Stock இனக்குழு netprice சேமிக்க தரவு உறுப்புகள் மற்றும் அதற்கு மதிப்பை கொடுக்கும் ஆக்கியை கொண்டிருக்கும். -Stock இனக்குழு, மொத்தவிலை (மொத்தவிலை என்பது 21% VAT உள்ளடக்கியது) கணக்கிட get-price என்ற செயற்கூறினை கொண்டிருக்கும்.

2. Stock இனக்குழுவை தருவித்துக் கொள்ளும் Notebook மற்றும் Book என்ற இரு இனக்குழுவை எழுதவும். - Notebook ன் மொத்தவிலை 21% வரியை சேர்த்து. - Books களுக்கு வரிவிலக்கு உள்ளதால் மொத்தவிலை, நிகரவிலையிலிருந்து மாறுபடாது, ஆகையால் நீ get Gross price என்ற செயற்கூறினை மறு வரையறை செய்ய வேண்டும்.

3. கீழ்கண்ட செயல்பாட்டினை செய்யும் நிரல் எழுதுக.

a. stock என்ற இனக்குழுவிற்கு 10 அணிகள் கொண்ட பொருள் உருவாக்கவும்.

b. Book மற்றும் Notebook என்ற இனக்குழுவிற்கு பொருள் உருவாக்கவும்.

c. பயனரிடம் book மற்றும் Notebook குறித்த தகவல்களை உள்ளீடு கேட்கவும்.

d. getgross pice என்ற செயற்கூறு ஒவ்வொரு வகைக்கும் சரியான வேலை செய்கிறதா என்பதை சோதித்து விடையை வெளியிடவும்.

துணை நூல்கள்:

(1) Object Oriented Programming with C++ (4th Edition), Dr. E. Balagurusamy, Mc.Graw Hills.

(2) The Complete Reference C++ (Forth Edition), Herbert Schildt. Mc.Graw Hills.

(3) Computer Science with C++ (A text book of CBSE XI and XII), SumitaArora, DhanpatRai& Co.

(4) A text book of CBSE XI and XII computer science by PreetiArora and Pinky Gupta.

(5) Computer Science with C++ Reeta shoo and Gagansahoo

(6) The C++ Programming Language,BjarneStroustrup

(7) C++ Primer (5th Edition) by S. B. Lippman, J. Lajoie

கணிப்பொறி நன்னெறி மற்றும் இணையப் பாதுகாப்பு

கற்றலின் நோக்கங்கள் :

- இணைய குற்றங்கள் பற்றி அறிந்து கொள்ளுதல்.
- இணைய உலகத்தில் இணையப் பாதுகாப்பு பற்றிய வழி காட்டுதல்கள் மற்றும் தேவைகள் பற்றி அறிதல்.
- இணையப் பாதுகாப்பு பற்றிய சிக்கல்கள் பற்றி அறிதல்.
- பிராக்ஸி சேவையகம் மற்றும் பயர்வால் செயல்பாடுகள் பற்றி அறிதல்.
- மறையாக்கம் மற்றும் குறியாக்கத்தின் அடிப்படை பற்றி கற்றல்.
- தகவல் தொழில்நுட்ப சட்டங்கள், விதிகள், செயல்படுத்துதல் பற்றி கற்றல்.

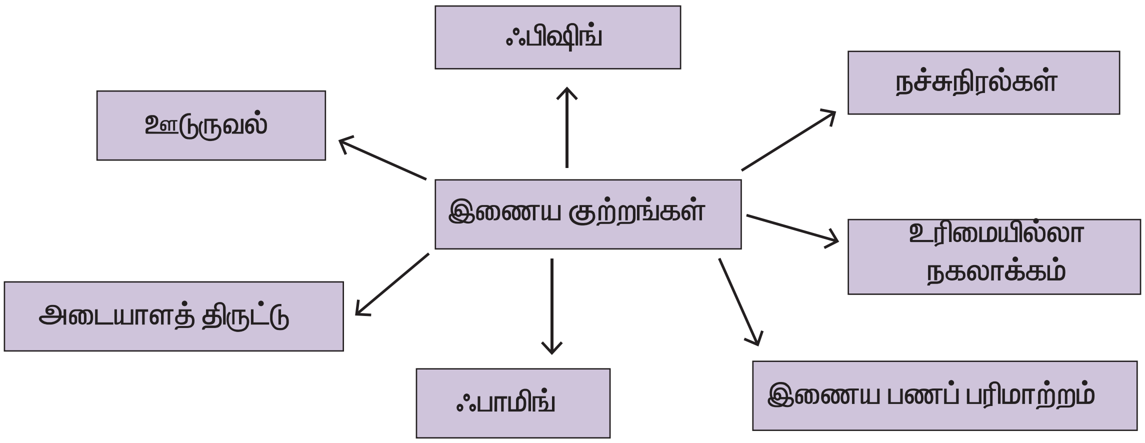
N2ISJ2

17.1 அறிமுகம்

இணையம் என்பது எளிதாக தொடர்பு கொள்ளக்கூடிய மற்றும் அனைவரும் பயன்படுத்தக்கூடிய சாதனமாக உள்ளது. தகவல் தொழில்நுட்பம் என்பது கணிப்பொறியின் வழியாக பரந்து விரிந்தது கணிப்பொறிகள், கைப்பேசிகள் மற்றும் இணையம் வழியாக தகவல் தொழில் நுட்பத்தின் நோக்கங்கள் பலவாக இருந்தாலும், அதை தவறாக பயன்படுத்தும் வாய்ப்புகள் உள்ளன.

கணிப்பொறி அமைப்பு என்பது பொதுவாக பாதிக்கப்பட கூடியது. அது தனிமனிதனின் அல்லது தொழில்களில், தினசரி வாழ்க்கையில் முக்கியப் பங்கு வகிக்கிறது.

மதிப்பு மிக்க தரவுகளை தவறாக பயன்படுத்துவரின் கைகளில் கிடைத்து விடாமல், சிறப்பு பாதுகாப்பு கொடுத்து பாதுகாக்க வேண்டும். ஆகவே, தரவுகள் பாதுகாக்கப்பட வேண்டும். இணைய குற்றங்கள் என்பது கணிப்பொறியிலும், வலைப்பின்னல்களிலும், ஈடுபடுத்தப்படுகிறது. இது வளர்ந்து வரும் சமூகத்தின் மீது குற்றவாளிகள், பொறுப்பற்ற தனிமனிதனால் வலைதளத்தை பயன்படுத்தி தாக்குதல்கள் ஏற்படலாம். இணைய குற்றம் என்பது குற்றமாகும் இதில் கணிப்பொறி மற்றும் வலைதளமும் அடங்கும். இது முக்கிய சவாலாக தகவல் தொழில்நுட்பம் பயன்படுத்துவோர் மீது உள்ளது. இ ை ண ய குற்றம், நேர்மை, பாதுகாப்பு மற்றும் வணிக அமைப்பின் உயிர் மீது அச்சுறுத்துவதாக உள்ளது.



படம் 17.1 பல வகையான இணைய குற்றங்கள்

நன்னெறி (ETHICS)

நன்னெறி என்பதன் அர்த்தம் எது தவறு ? மற்றும் எது சரி? இது கணிப்பொறி யார் பயன்படுத்துகிறார்களோ, அவர்களின் தார்மீக கொள்கையின் தொகுப்பு ஆகும். சரியான நடவடிக்கையை பின்பற்றி, நெறிமுறைகளை பயன்படுத்தி வருவது நன்னெறி ஆகும்.

அறநெறி என்பது சமூகத்தில் (Morals) உள்ள நல்லவை, கெட்டவைகளை ஏற்று நடப்பது ஆகும். இன்றைய இணைய உலகில் சில தர நிலைகள் உள்ளன.

- திருட்டு மென்பொருளை பயன்படுத்தாமல் இருப்பது.
- அடுத்த பயனரின் கணக்கை அனுமதியின்றி பயன்படுத்தாமல் இருப்பது.
- அடுத்தவரின் கடவுச்சொல்லை திருடாமல் இருப்பது.
- ஊடுருவல் செய்யாமல் இருப்பது.

கணிப்பொறியின் நன்னெறியின் முக்கிய பிரச்சினைகள், இணைய சேவை பயன்படுத்துவோரின் தனிப்பட்ட தகவல்கள், பதிப்புரிமம் மற்றும் உரிமம் பெற்ற பொருளடக்கத்தை வெளியிடுதல்.

கணிப்பொறி நன்னெறி (COMPUTER ETHICS)

இணையத்தின் உதவியால் உலகமானது உலக கிராமமாக தற்போது உள்ளது. தனிமனிதன், வெவ்வேறு வணிக நிறுவனங்களின் பெயர்கள் மிகவும் பிரபலமானதாக இணையத்தின் உதவியால் உள்ளது. மின் வணிகம் என்பது மிக பிரபலமானதாக விரிந்த பரப்பில் வேகமானதாக உள்ளது.

கணிப்பொறி நன்னெறி செயல்முறை என்பது மதிப்புகள், நடைமுறைகள் முதலியவற்றை கையாள்கிறது. நுகர்வோர் செயல்முறையை, கணிப்பொறி தொழில்நுட்பத்தின் மாற்றம் அது தொடர்புடைய ஒழுக்கத்தை, சேதமோ அல்லது நீதி நெறியை மீறியதாகவோ மற்றும் தனிமனித நிறுவனம் ஆகியவற்றில் நீதி நெறி மீறாமல் நிர்வகிப்பதாகும். இது மனித சமூகத்தின் செயற்கை நுண்ணறிவின் (AI) மீது எவ்வாறு தாக்கத்தை ஏற்படுத்துகிறது என்ற விவாதத்தையும் ஊக்குவிக்கிறது.

நன்னெறியின் வழிகாட்டுதல்கள் (GUIDELINES OF ETHICS)

பொதுவாக, பின்வரும் வழிகாட்டுதல்கள் கணிப்பொறி பயன்படுத்துவர்கள் கவனத்தில் கொள்ள வேண்டும்.

1. நேர்மை (Honesty) : இணையத்தை பயன்படுத்தும் பயனர் உண்மையுள்ளவராக இருத்தல்.

Chapter 17.indd 306 03/09/18 5:52 PM

2. நம்பிக்கை (**Confidentiality**): பயனர் அங்கீகரிக்கப்படாதவர்களிடம் முக்கிய தகவல்களை பரிமாற்றம் செய்யாமல் இருத்தல்.

3. மரியாதை (**Respect**) : மற்ற பயனருக்கு உள்ள தனி உரிமைக்கு உரிய மரியாதையை ஒவ்வொரு பயனரும் கொடுத்தல்.

4. தொழில்முறை (**Professionalism**) : தொழில்முறையில் ஒவ்வொரு பயனரும் தொழில் முறை நடத்தையுடன் இருத்தல்.

5. சட்டத்திற்கு கீழ்படிதல் (**The Law**): சட்டத்திற்கு கணிப்பொறி பயன்பாடுகளின் கீழ்படிதல் போன்று இணைய சட்டத்திற்கும் கீழ்படிதல்.

6. பொறுப்பு (**Responsibility**) : பயனர் ஒவ்வொருவரும், ஒவ்வொரு செயலுக்கும் உடைமையாளராக பொறுப்பேற்றுக் கொள்ளுதல்.

உங்களுக்குத் தெரியுமா?

நன்னெறி என்பது சமுதாயத்தின் தனிமனித நடத்தைகள் அறநெறி கொள்கையின் தொகுதிகளால் ஆனது மற்றும் அறநெறி கொள்கையின் தொகுப்பு கணிப்பொறி பயன்படுத்தும் பயனரை கட்டுப்படுத்தும்.

17.2 நன்னெறியின் பிரச்சினைகள் (ETHICAL ISSUES)

நன்னெறி பிரச்சினை என்பது, பிரச்சினை அல்லது தனி மனிதனுக்கோ அல்லது நிறுவனத்திற்கோ ஏற்படும். சரி (நன்னெறி) அல்லது தவறு (நன்னெறி அல்லாதது) இவற்றின் ஒன்றை தேர்வு செய்யும் முறை ஆகும்.

இந்த பிரச்சினை ஆனது தீர்க்கப்பட்ட அல்லது தீர்க்கப்படாத, சமூகத்தில் உள்ள நேர்மறையான அணுகுமுறையை சேர்க்கப்படும். சில பொதுவான நன்னெறி பிரச்சினைகள் கீழே பட்டியலிடப்பட்டுள்ளது.

- இணைய குற்றம் (cyber crime)
- மென்பொருள் உரிமையில்லா நகலாக்கம் (software piracy)
- அங்கீகரிக்கப்படாத அணுகுதல் (un Authorized access)
- ஹேக்கிங் (Hacking)
- கணிப்பொறியை பயன்படுத்தி மோசடி செய்தல் (Use of computers to commit fraud)
- நச்சு நரல்(Virus) மூலம் நாசவேலை
- கணிப்பொறி மூலம் தவறான கூற்று உருவாக்குதல்.

இணைய குற்றம் (Cyber Crime)

இணைய குற்றம் என்பது அறிவுசார் வெள்ளைக்காலர் குற்றமாகும். இந்த குற்றங்களை செய்வோர் பொதுவாக கணிப்பொறியை திறன்பட இயக்குபவராக இருப்பார்கள்.

உதாரணமாக – சட்ட விரோத பண பரிவர்த்தனை இணையத்தின் வழியாக நடைபெறுவதை உதாரணமாக கூறலாம். கணிப்பொறி குற்றங்களில் சில உதாரணமாக கொடுக்கப்பட்டுள்ளது.

Chapter 17.indd 307 03/09/18 5:52 PM

அட்டவணை 17.1

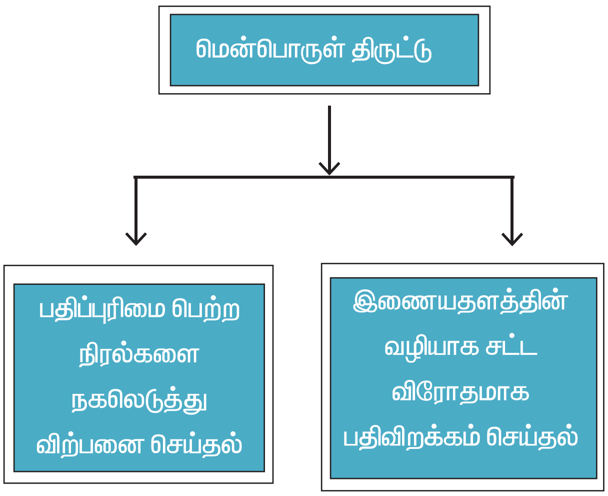
குற்றம்	செயல்பாடுகள்
இணைய தீவிரவாதம் (Cyber terrorism)	அச்சுறுத்துதல், மிரட்டல் மற்றும் பிளாக்மெயில் மூலம் ஒருவரிடமோ அல்லது வணிகரிடமோ திருடுதல்.
இணையத் தொந்தரவு (Cyber stalking)	இணையத் தொந்தரவு மூலம் திருடுதல்
தீம்பொருள் (Malware)	இணையவழி தொந்தரவு பல்வேறு செயல்களான திருடுதல், மறையாக்கம் அல்லது முக்கியமான தரவுகளை நீக்கம் செய்தல், எச்சரிக்கை அல்லது கணிப்பொறி செயல்பாடுகளை நடத்துதல், செயல்பாடுகளை அனுமதி இல்லாமல் கண்காணித்தல்.
சேவை தாக்குதல்களின் மறுப்பு (Denial of service attack)	போலி கோரிக்கைகள் மூலம் அளவுக்கதிகமான கணிப்பொறி அமைப்பினால் சாதாரண சட்டவிரோத கோரிக்கைகள் பணியாற்ற இயலாத நிலை.
மோசடி (Fraud)	அங்கீகாரம் இல்லாமல் பண பரிமாற்றம். தரவுகளை கையாளும் போது உதாரணமாக வங்கி பதிவுகளை மாற்றுவதன் மூலம் பண பரிவர்த்தனையில் மோசடி செய்தல்.
அரண் உடைத்தல் (Harvesting)	சட்டவிரோதமாக அடுத்த பயனரின் பயனர் பெயர் மற்றும் கடவுச் சொல்லை சேகரித்து பயனரின் கணக்குகளில் நுழைந்து பயனடைதல்.
அடையாளத் திருட்டு (Identity theft)	நிதி ஆதாயத்திற்காக, தனி நபரின் அடையாளத்தை குற்றவாளிகள் பயன்படுத்துதல்.
அறிவுசார் சொத்து திருட்டு (Intellectual property theft)	ஒரு நிறுவனத்தால், தனி நபரால் உருவாக்கப்பட்ட நடைமுறை அல்லது கருத்தியல் தகவலை திருடுதல்)
சலாமி ஸ்லைசிங் (Salami slicing)	இணைய பண பரிவர்த்தனையில் சிறிய அளவாக பணம் திருடுதல்..
ஊழல் (Scam)	உண்மை இல்லாத ஒன்றை, மக்களை நம்ப வைத்து ஏமாற்றுவது.
ஸ்பேம் (Spam)	அறியப்படாத மூலத்திலிருந்து பெறுபவர் அறியப்பட்ட ஆதாரத்தை அனுப்பி தீங்கிழைக்கும் நடைமுறையாகும்..
ஏமாற்றுதல் (Spoofing)	தேவையற்ற மின்னஞ்சலை அதிக எண்ணிக்கையில் இணைய தள பயனர்களுக்கு அனுப்புதல்.

மென்பொருள் திருட்டு (SOFTWARE PIRACY)

மென்பொருள் திருட்டு என்பது ஒரு தனிப்பட்ட அல்லது ஒரு நிறுவனத்தால் முதலில் உருவாக்கப்பட்ட மென்பொருளை பதிப்புரிமை பெறாமல், சட்ட விரோதமாக குறியீடுகள், தகவல்கள், நிரல்கள் மற்றும் பிற தகவல்களை திருடுதல். அங்கீகாரம் இல்லாமல், நகல்களின் பிரதிகளை உருவாக்கி இந்த தரவை சொந்த நலனுக்காக, அல்லது வணிக இலாபத்திற்காக பயன்படுத்துவது ஆகும்.

Chapter 17.indd 308 03/09/18 5:52 PM

எளிமையான சொற்களில் மென்பொருள் திருட்டு என்பது "மென்பொருள்களின் அங்கீகரிக்கப்படாத நகல்" ஆகும்.



படம் 17.2- மென்பொருள் திருட்டின் விளக்கப்படம்

பெரும்பாலான மென்பொருள்கள் ஒரே கணிப்பொறியில் மட்டுமே பயன்படுத்த அனுமதிக்கப்படுகின்றன அல்லது ஒரே நேரத்தில் ஒரு பயனரால் மட்டுமே பயன்படுத்தப்படுகின்றன. ஒரு பயனர் மென்பொருளை வாங்கும் போது அதன் மென்பொருள் உரிமம் பெற்ற பயனராகிறார்.

காப்புரிமை பயன்பாட்டிற்கான நகல்களில் பிரதிகள் தயாரிக்க அனுமதிக்கப்படுவர். நகல்களை மற்றொருவருக்கு விநியோகம் செய்வது சட்ட விரோதமானது.

மென்பொருள் திருட்டிற்கு முற்றிலும் மாறுபட்ட அணுகுமுறை பகிர்மான மென்பொருள் என அழைக்கப்படுகின்றன. நகலெடுப்பதில் இருந்து மக்களை தடுக்க முயற்சிப்பதால் பயனில்லை. மாறாக மக்களுக்கு நேர்மையை உணரச்செய்யலாம்.

ஆசிரியர்கள் நேரடியாக ஒரு பதிவு கட்டணத்தை செலுத்தி, பயனர்களும், சக ஊழியர்களும், நிரல்களை நகலெடுக்க ஊக்கப்படுத்தப்படுகிறது. சட்ட விரோதமாக பொதுமக்களுக்கு கிடைக்கக்கூடிய வணிக நிகழ்ச்சிகள் பெரும்பாலும் வார்ஸ்கள் என்று அழைக்கப்படுகின்றன.

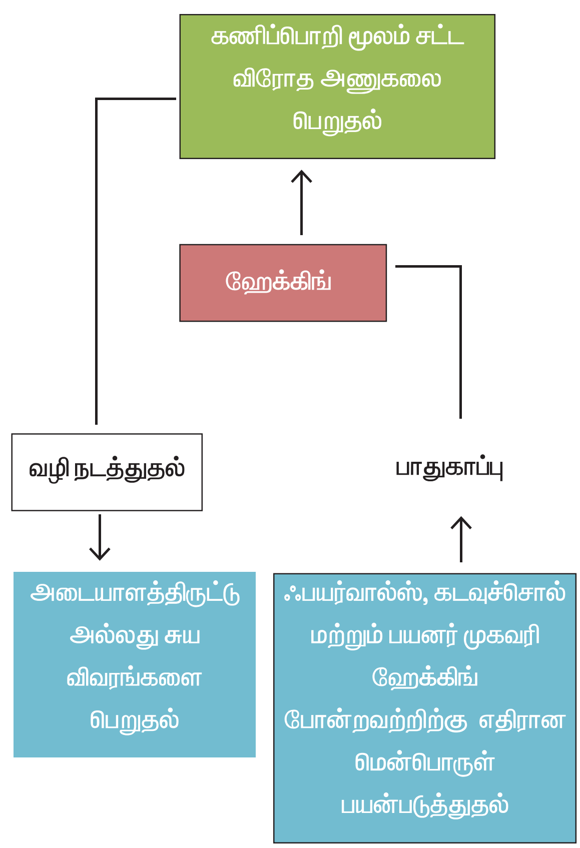
அங்கீகரிக்கப்படாத அனுமதி

அங்கீகாரமற்ற அணுகல் என்பது ஒரு வலைதளம், நிரல், சேவையகம் அல்லது பிற முறைமைக்கான அணுகல் முறையான பயனர் கணக்கை மீறுவதன் மூலம் அது அங்கீகரிக்கப்படாத அணுகல் எனப்படும்.

அங்கீகரிக்கப்படாத அணுகுதலை தடுக்க பயர்வால்கள், இண்டெர்வியூ டிடெக்டிக் சிஸ்டம்ஸ், CITS, நச்சு நிரல் மற்றும் உள்ளடக்க வருடிகள் பயன்படுத்தப்படுகின்றன.

ஹேக்கிங்

ஹேக்கிங் என்பது ஒரு கணிப்பொறியின் உரிமையாளரின் அனுமதி இல்லாமல், தனிப்பட்ட தரவு அல்லது கடவுச் சொல்லை குற்றம் சார்ந்த நடவடிக்கையாகவோ அல்லது பொழுது போக்கிற்காகவோ திருடுதல். இத்தகைய நெறிமுறை ஹேக்கிங் மட்டுமே கட்டுப்படுத்தப்பட்ட சோதனைகள் என நடைமுறைப்படுத்தப்பட வேண்டும்.



படம் 17.3 ஹேக்கிங் வரைபடம்

Chapter 17.indd 309 03/09/18 5:52 PM

கிராக்கிங்

ஒரு திட்டத்தின் மூலத்தை எடிட் செய்து குறியீட்டை பயன்படுத்தி மாற்றலாம். கிராக் (ஒரு கருப்பு தொப்பி) அல்லது இருண்ட பக்க ஷேக்கர் என்றும் அழைக்கப்படுகின்றது.)

ஒரு தீய, குற்றவியல் ஷேக்கர் "கிராக்கிங்" என்றால் திருட்டு, ஊழல் அல்லது சட்ட விரோதமாக தரவைப் பார்க்க கணிப்பொறிகளை பயன்படுத்துதல் ஆகும்.

மென்பொருள் சிதைவு என்பது மிகவும் அடிக்கடி ஏற்படுத்தப்படும் சிதைவு வகை ஆகும். கடவுச்சொல் விரிசல் என்று மற்றொரு வகை உள்ளது. இது கடவுச் சொற்களை அழிக்கப்பயன்படுகிறது. தானியங்கு திட்ட நிரல்களை பயன்படுத்தி கடவுச் சொல் கிராக் செய்வது அல்லது கைமுறையிலும் செய்யலாம். கிராக்கிங் என்பது ஒரு சுவாரசியமான உண்மை, சமூக கட்டமைப்பு ஆகும். மனிதனுடைய பலவினத்தைப் பயன்படுத்தி, கடவுச் சொற்கள் மட்டும் தகவல்களை பெறும் வழியாகும். இது மென்பொருள் அல்ல தகவல்களைப் பெறுவதற்கு தொலைபேசியை பயன்படுத்தலாம். உங்கள் நண்பர்களாக இருப்பதுடன் இணைய ரிலே அரட்டை (IRC) அல்லது நல்லவராக உங்களுடன் பேசி தகவல்களை பெறலாம். மின்னஞ்சல் அவர்களுக்கு ஒரு அதிகாரபூர்வ மின்னஞ்சலாக, வங்கி அல்லது பிற அதிகார பூர்வ நிறுவனங்களின் மின்னஞ்சல் போல் தோன்றலாம்.

உங்களின் தனிப்பட்ட தகவல்களை அவர்கள் கண்டுபிடித்து கடவுச்சொல்லை யூகிக்க முயற்சி செய்கிறார்கள். தனிப்பட்ட நன்மைக்காக அவர்களே கண்டுபிடித்து அவற்றை சுரண்டுவதில் உள்ள பாதிப்புகள் பற்றி அறிந்து கொள்ளலாம்.

17.3 இணையப் பாதுகாப்பு மற்றும் அச்சுறுத்தல்கள்

ஒரு கணிப்பொறிக்கு கணிசமான சேதத்தை ஏற்படுத்தவோ அல்லது ஒரு நபரிடமிருந்து அல்லது ஒரு நிறுவனத்திடமிருந்து முக்கியத் தகவல்களை திருடுவதாகவோ, இணைய தாக்குதல்கள் முதன்மையாக தொடங்கப்பட்டது. இணையப் பாதுகாப்பு என்பது பல்வேறு தொழில்நுட்பங்கள் செயல்முறைகள் மற்றும் நடவடிக்கைகள் ஆகியவற்றின் தொகுப்பாகும். இது இணைய தாக்குதல்களின் ஆபத்தை குறைக்கும் மற்றும் நிறுவனங்கள் மற்றும் தனிநபர்கள் பாதுகாக்கும் கணிப்பொறி சார்ந்த அச்சுறுத்தல்களில் இருந்து பாதுகாக்கும்.

இணையத் தாக்குதல்களின் வகைகள்

மென்பொருள் சட்ட விரோதமான அணுகல் மற்றும் சேதம் விளைவிக்கும் வகையில் வடிவமைக்கப்பட்ட ஒரு வகை மென்பொருள் ஆகும். பல வகையான இணைய தாக்குதல்கள் மற்றும் அவற்றின் செயல்பாடுகள் ஆகியவை அட்டவணை 17.2 ல் கொடுக்கப்பட்டுள்ளன..

அட்டவணை 17.2 – இணையத் தாக்குதல்கள் மற்றும் செயல்பாடுகள்

வ.எண்	இணையத் தாக்குதல்கள்	செயல்பாடுகள்
1.	நச்சு நிரல்	ஒரு நச்சு நிரல் என்பது கணிப்பொறி குறியீட்டின் ஒரு சிறிய பகுதி ஆகும். அது தன்னை மீண்டும் மீண்டும் ஒரு கணிப்பொறியில் இருந்து மற்றொரு கணிப்பொறிக்கு கோப்புடன் இணைக்கும் வகையில் பரவுகிறது. பொதுவான நச்சு நிரல் ட்ரோஜன் ஆகும். ஒரு ட்ரோஜன் நச்சு நிரல் என்பது ஒரு செயல்பாடு. எடுத்துக்காட்டாக – நச்சு நிரல் நீக்கம் ஒரு எடுத்துக்காட்டாக கூறலாம். உண்மையில் நச்சு நிரல் செயல்படுத்தப்படும் போது தீங்கிழைக்கும் செயல்களை செய்கிறது.

2.	வார்ம்ஸ்	வார்ம்ஸ் என்பது சுயமாக திரும்ப திரும்ப வந்து இணைத்துக் கொள்ள நிரல்கள் தேவை இல்லை. வார்ம்ஸ் தொடர்ந்து பாதிப்புக்குள்ளாகி பலவீனங்களை கண்டுபிடித்து வார்ம்ஸின் நிரலாலருக்கு தெரிவிக்கிறது.
3.	ஸ்பைவேர்	கணிப்பொறியின் இணைப்புக்களை திறக்கும் போது தானாகவே கணிப்பொறியில் நிறுவப்படலாம். இணைப்புகளில் கிளிக் செய்யும் போதும் பாதிக்கப்பட்ட மென்பொருளை பதிவிறக்கம் செய்வதன் மூலமும் ஸ்பைவேர் நிறுவப்படலாம்..
4.	ரேன்சம்வேர்	ஒரு கணிப்பொறியில் இணைய தாக்குதல்களில் தொடங்குவதற்கு பிறகு பணம் கோரி தீங்கு இழைக்கத் திட்டமிடுதல். இந்த தீம்பொருள் குற்றவாளி களுக்கிடையே பெருகிய முறையில் பிரபலமடைந்து ஒவ்வொரு வருடமும் நிறுவனங்களுக்கு மில்லியன் கணக்கான செலவுகளை ஏற்படுத்துகிறது..

இணையப் பாதுகாப்பு (threats)

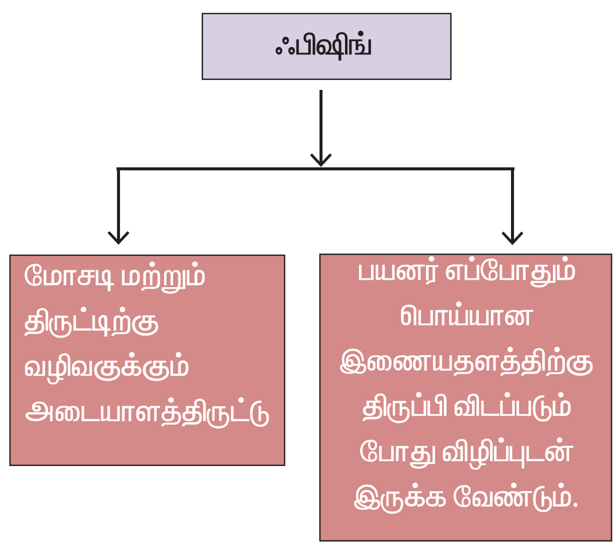
சமீபத்திய வருடங்களாக பாதுகாப்பு அமைப்புகளில் உள்ள பலவீனங்கள் காரணமாக தனிநபர்கள் மற்றும் நிறுவனங்கள் பெரும்பாலான பிரச்சினைகளை எதிர்கொள்கிறது. பல்வேறு வகையான இணையப்பாதுகாப்பு அச்சுறுத்தல்கள் கீழே வகைப்படுத்தப்பட்டுள்ளன.

சமூக கட்டமைப்பு

சமூக கட்டமைப்பு ஒரு நபரின் பலவீனத்தை தவறாக பயன்படுத்துதல் தீங்கிழைக்கும் இணைப்புகளை கிளிக் செய்வதன் மூலமும் அல்லது நுணுக்கங்கள் மூலம் கணிப்பொறியை அணுகுவதாலும் பெறப்படுகிறது.ஃபிஷிங் மற்றும் ஃபார்மிங் சமூக கட்டமைப்புகளுக்கு உதாரணங்கள் ஆகும்.

17.3.1. ஃபிஷிங்

ஃபிஷிங் என்பது கணிப்பொறி குற்றத்தின் ஒரு வகை ஆகும். கடவுச்சொல் மற்றும் கிரெடிட் கார்டு எண்கள் உள்ளீட்ட பயனர் தரவை திருடுவதற்கு பயன்படுத்தப்படும். ஒரு மின்னஞ்சலில் ஒரு உடனடி உரைச் செய்திப்பெட்டியை திறக்கும் போது அது பாதிப்புக்குள்ளாகி தீங்கிழைக்கக்கூடிய இணைப்புகள் அல்லது இணைப்புகளை விநியோகிக்க ஃபிஷிங் பயன்படுகிறது. ஒரு பாதிக்கப்பட்டவரிடமிருந்து முக்கிய உள்நுழைவு சான்றுகளை பிரித்தெடுத்தல் உட்பட பலவிதமான செயல்பாடுகளை செய்ய முடியும்.



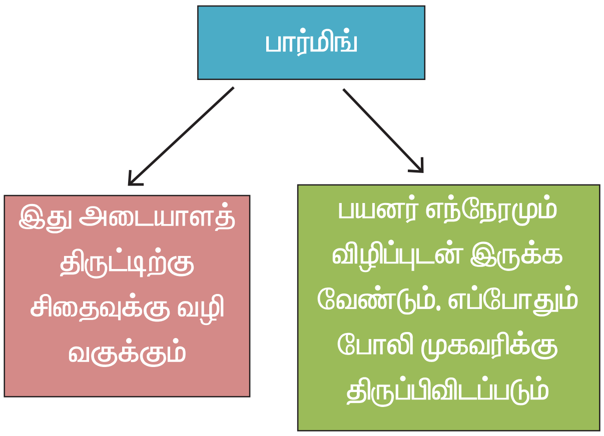
படம் 17.4 ஃபிஷிங் வரைபடம்

17.3.2. பார்மிங் (Pharming)

பார்மிங் என்பது ஒரு தனிப்பட்ட கணிப்பொறியின் அல்லது சேவையகத்தின் தீங்கிழைக்கும் குறியீடு உள்ள மோசடியான ஒரு நடைமுறை ஆகும்.

பயனரின் அனுமதி இல்லாமல் தவறான வலை தளங்களை தவறாக வழி நடத்துகிறது. பயனரை மோசடி செய்ய ஷேக்கர்கள் முயற்சி செய்து வலைதளத்தின் போக்கை திசை திருப்பும் ஒரு இணையத் தாக்குதல் ஆகும்.

Chapter 17.indd 311
03/09/18 5:52 PM

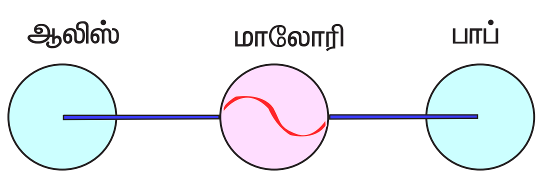


படம் 17.5 பார்மிங் வரைபடம்

ஒரு நபர் மீது தாக்குதல் (Man In The Middle (MITM)) :

ஒரு நபர் மீது தாக்குதல் (MITM மற்றும்) என்பது இரகசியமாக தொடர்பு ஏற்படுத்தி தாக்குதல், மேலும் ஒருவரோடு ஒருவர் நேரிடையாக தொடர்பு கொள்வதாக நம்புகிற இரு கட்சிகளிடையே தொடர்பை மாற்றியமைத்து, தாக்குதல் தொடுப்பது ஆகும்.

எடுத்துக்காட்டு பாப் உடன் தொடர்பு கொள்ள ஆலிஸ் விரும்புகிறார் என கருதினால் இடையில் பல்வேறு உரையாடலை இடைமறித்து, பாப் தவறான செய்தி பெறுவது போல் செய்வது ஆகும்.



படம் 17.6 – ஒரு நபர் மீது தாக்குதல்

குக்கிகள் (Cookies)

ஒரு குக்கி (இணைய குக்கி, HTTP குக்கி, வெற்று குக்கி, உலவி குக்கி அல்லது சாதாரண குக்கி) என்பது வலை தளத்திலிருந்து அனுப்பப்பட்ட ஒரு சிறிய துண்டு தரவு மற்றும் பயனரின் இணையமானது அனைத்து வலை தளங்களின் ஒரு இணைய தளத்தில் இணைய தள அங்காடியில் சேர்க்கப்பட்ட பொருட்கள் போன்றவை ஆகும்.

பயனரின் பயன்பாட்டால் பொத்தானை கிளிக் செய்து உள் நுழைதல் அல்லது பதிவு செய்தல், வலைதளங்களுக்கான நம்பகமான கருவியாக வடிவமைக்கப்பட்டுள்ளது. பயனரின் பழைய பயர்வால், முகவரிகள், கடவுச் சொற்கள் மற்றும் கடன் அட்டை எண்கள் போன்ற பிரபலங்கள் உள்ளிட்ட பயனர் தன்னிச்சையின் தகவல்களை நினைவில் வைத்து பயன்படுத்தலாம். குக்கி தரவு இடைமறிக்கப்பட்டால் பாதுகாப்பு கண்ணோட்டத்தில் இருந்து, பயனர் குக்கி தகவலை அறிந்து. தவறாக பயன்படுத்தவும் வாய்ப்பு உள்ளது. பின்வரும் காரணங்களுக்காக வலைதளங்கள் பொதுவாக குக்கிகளை பயன்படுத்துகின்றன.

- இணைய தளத்திற்கு சென்று பார்த்தவர்கள் பற்றிய விவரங்களை சேகரிக்க, இந்த இணைய தளத்தை பெரும்பாலும், எத்தனை பார்வையாளர்கள் பார்வையிட்டனர் எவ்வளவு நேரம் பார்வையிட்டார்கள் என்பதையும் கண்காணிக்கலாம்.
- இது வலை தளத்தின் பயனரின் அனுபவத்தை தனிப்பயனாக்க உதவுகிறது.
- குக்கிகள் பயனர் பற்றியத் தனிப்பட்ட தகவல்களை சேமித்து வைக்க உதவும்.
- இதன் மூலம் ஒரு பயனர் தளத்திற்கு திரும்பும் போது அதிக அனுபவங்கள் பெறப்படுகின்றன.

இணைய தளத்தில் எப்போதாவது திரும்பி சென்றால், உங்கள் பெயர் மர்மமாக திரையில் தோன்றும், முந்தைய வருகையின் காரணமாக உங்கள் பெயரை தளத்தில் கொடுத்து இருந்தீர்கள் என்றால் அதே குக்கியில் சேமிக்கப்பட்டுள்ளதற்கு ஒரு சிறந்த உதாரணம்.

இணையதள அங்காடி தளத்தில், ஏற்கனவே பொருள்கள் வாங்கியதன் அடிப்படையில் பயனர்களுக்கு பரிந்துரைக்கும் வழியாகும். விளம்பரங்களை கண்காணிக்கவும் குக்கிகள் பயன்படுத்துகின்றன. குக்கிகள் கணிப்பொறியில் தீமை செய்யவில்லை அவை எந்த நேரத்திலும் நீக்கப்பட கூடிய உரை கோப்புகள் ஆகும்.

Chapter 17.indd 312 03/09/18 5:52 PM

நச்சுநிரல்கள் பரப்புவதற்கு குக்கிகளை பயன்படுத்த இயலாது. மேலும், அவை உங்கள் நிலைப்பாட்டை அணுக இயலாது. குக்கியின் அம்சமானது வெளிப்படையாக உங்கள் உலவியில் நிறுத்தப்படாவிட்டால் கடன் அட்டை தகவலை உள்ளடக்கிய வலைத் தளத்திற்கு வழங்கப்படும்.

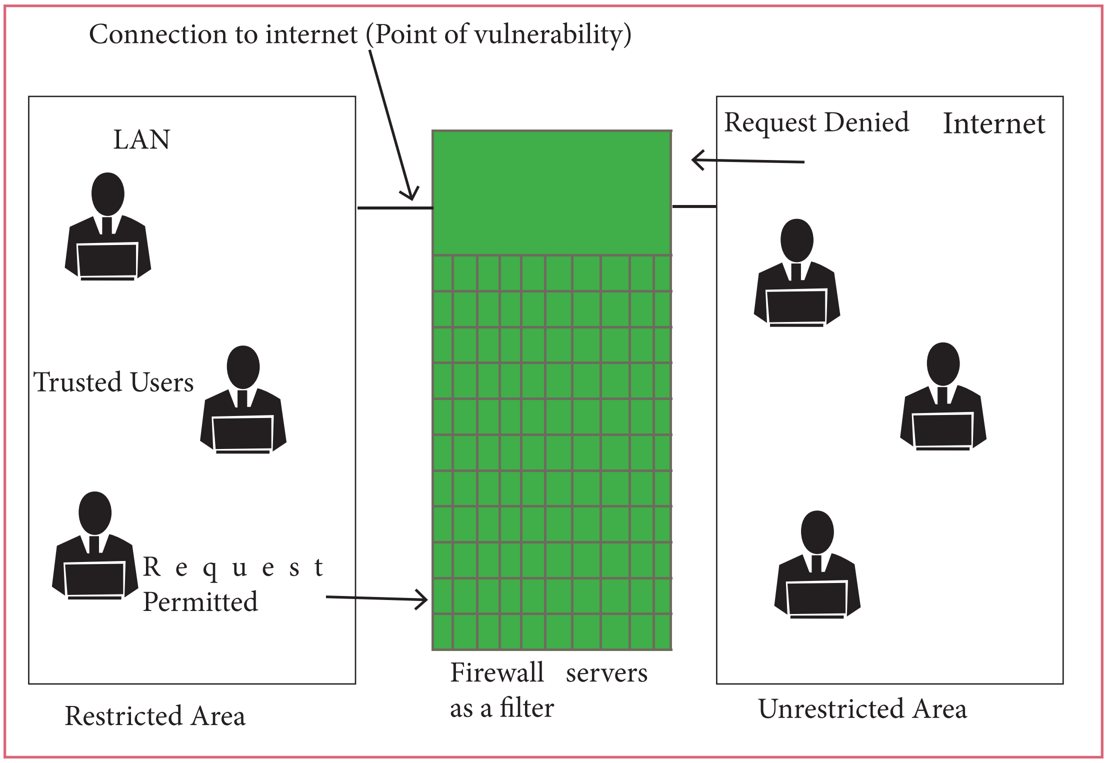
எந்த ஒரு தனிப்பட்ட தகவலும் பெரும்பாலும் குக்கியில் சேமிக்கப்பட்டாலும் இது குக்கிகளின் தனியுரிமையை அச்சுறுத்தும் வழியாகும்.

ஃபயர்வால் மற்றும் பதிலாள் சேவையகங்கள்

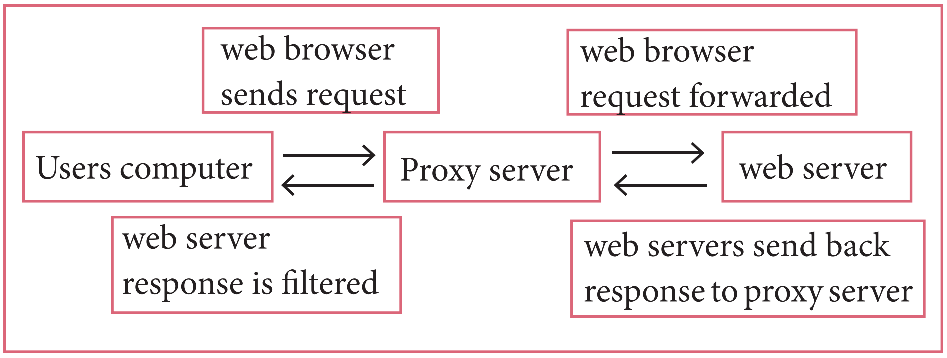
ஃபயர்வால் மற்றும் பதிலாள் சேவையகங்கள் என்பது கணிப்பொறி வலையமைப்பு பாதுகாப்பு அடிப்படை அமைப்பாகும். பாதுகாப்பு அடிப்படையில் உள்வரும் மற்றும் வெளிச் செல்லும் வலையமைப்பு போக்குவரத்து போன்றவற்றை கண்காணித்து கட்டுப்படுத்துகிறது. ஃபயர்வால் பொதுவாக நம்பகமாக உள்ள கணிப்பொறி வலைதளம் மற்றும் வலைப்பின்னலுக்கு வெளியே ஒப்படைக்கப்பட்ட கணிப்பொறிக்கும் இடையே ஒரு பிளாக் அமைகிறது. இணைய அடிப்படையிலான அல்லது ஷோஸ்டை அடிப்படையாக கொண்டு வகைப்படுத்தப்படுகின்றன. வலைப்பின்னல் அடிப்படையிலான ஃபயர்வால்கள் LAN, WAN ஆகியவற்றில் கணிப்பொறி நுழைவாயில் வைக்கப்பட்டுள்ளன.

ஃபயர்வால் என்பது இயங்கு தளத்தின் ஒரு பகுதியாக இறுதி பாதுகாப்பு அல்லது பாதுகாப்பு போன்ற ஏஜென்ட் பயன்பாட்டாக இருக்கலாம். ஒவ்வொன்றிலும் நன்மைகள் மற்றும் தீமைகள் உள்ளன. இருப்பினும் ஒவ்வொன்றும் அடுக்கு பாதுகாப்புக்கும் ஒரு பங்கு உள்ளது. பயர்வால்களின் வகைகள் மாறுபடுகின்றன.

ஒரு பிராக்ஸி சேவையகம். இறுதி பயனர்களுக்கும், வலை சேவையகத்திற்கும், இடையில் இடைத்தரகராக செயல்படுகின்றன. கோப்பு இணைப்பு, வலைப்பக்கம் அல்லது வேகமான சேவையகத்திலிருந்து கிடைக்கும் புற ஆதாரங்கள் போன்ற சில இணையைத் தேடுகிறது. பிராக்ஸி சேவையகம் கோரிக்கையை ஆராய்கிறது. நம்பகத்தன்மையை ஆராய்ந்து அதன்படி கோரிக்கை வழங்கப்படுகிறது. நம்பகத்தன்மையை ஆராய்ந்து பிராக்ஸி சேவையகங்கள் பொதுவாக அடிக்கடி பார்வையிடும் தள முகவரிகள் அதன் தற்காலிக சேமிப்பில் மேம்பட்ட பதிலளிப்பு நேரத்திற்கு வழிவகுக்கும்.



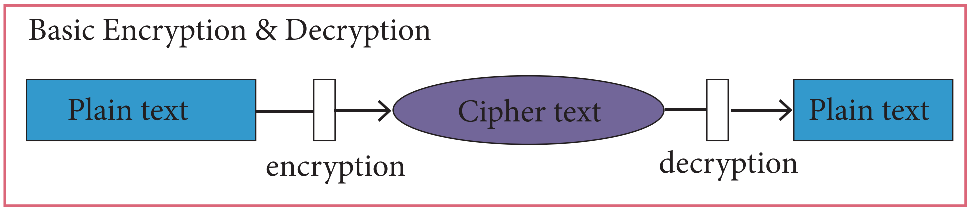
படம் 17.7 ஃபயர் சேவையகம்



படம் 17.8 பிராக்ஸி சேவையகத்தின் செயல்பாட்டை படம் விளக்குகிறது.

குறியாக்கம் மற்றும் மறை குறியாக்கம்

குறியாக்கம் மற்றும் மறை குறியாக்கம் என்பது அங்கீகரிக்கப்பட்ட நபர்கள் மட்டுமே தகவலை அணுக முடியும் என்ற இரகசியத்தை உறுதிப்படுத்தும். செயல்முறைகள் குறியாக்கமானது எளிய உரைத் தரவு சீரற்ற மற்றும் சிக்கனமான தரவுகளாக (சைஃபி உரை) என அழைக்கப்படுகின்றன. குறியாக்கம் மற்றும் மறையாக்கம் ஆகிய இரண்டும் குறியாக்கத்தை கொண்டு செய்யப்படுகிறது. குறியாக்கம் என்பது தகவலின் துண்டுகளால் உருவாக்கப்பட்டது.



படம் 17.9 குறியாக்கம் மற்றும் மறை குறியாக்கம்

குறியாக்கம் என்பது இரகசிய குறியீடுகளை பயன்படுத்தும் அரசாங்கம் மற்றும் இராணுவத்தில் தொடர்பு கொள்ள பயன்படுத்தும் இரகசிய மொழியாக பயன்படுகிறது. தகவலை பாதுகாப்புடன் தொடர்பு கொள்ள பயன்படுகிறது. உதாரணமாக, தரவுகள் வலைதளம், தொலைபேசி கம்பியில்லா நுண்ணிய தொலைபேசி, புளூடூத் மற்றும் தானியங்கி கருவிகள் போன்றவற்றின் மூலம் பரிமாற்றம் செய்யப்படுகிறது. அண்மை ஆண்டுகளில் தகவல் தொடர்பில் குறிப்பிடத்தக்க குறுக்கீடுகள் உள்ளதாக எண்ணிலடங்கா குற்றச்சாட்டுக்கள் உள்ளன.

வலைதள போக்குவரத்திற்கு எதிராக பாதுகாப்பதாக வலைதளம் முழுவதும் அனுப்பப்படும் போது தரவும் குறியாக்கம் செய்யப்பட வேண்டும்.

குறியாக்கத்தின் வகைகள்:

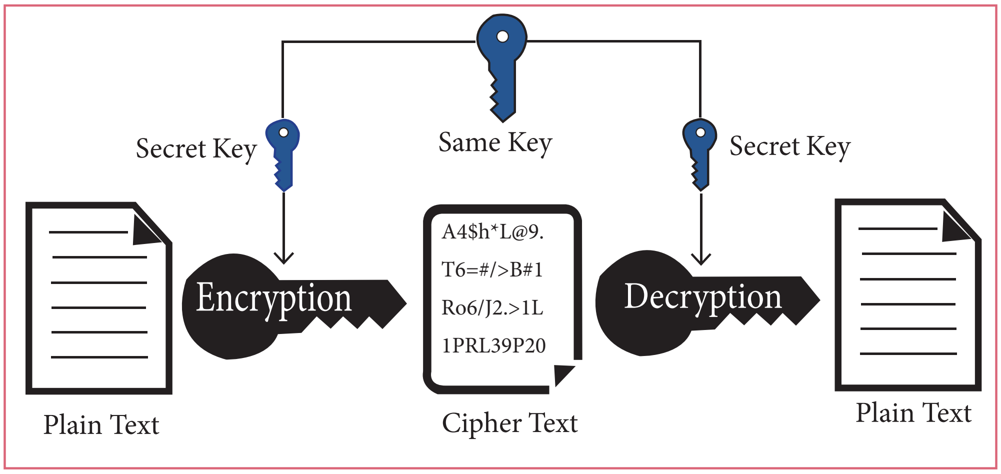
குறியாக்கத்தின் இரண்டு வகை குறியாக்கத் திட்டங்கள் கீழே பட்டியலிடப்பட்டுள்ளது.

1. சமச்சீர் குறியாக்கம்

2. பொது குறியாக்கம்.

சமச்சீர் குறியாக்கம், பொது குறியாக்கம், முக்கிய குறியாக்கம் மற்றும் சமச்சீர் மறை குறியாக்கம் இரண்டிற்கும் ஒரே விசையை பயன்படுத்துவதற்கான ஒரு நுட்பமாகும். சமச்சீர் விசை குறியாக்கத்தின் முக்கிய குறைபாடு என்னவென்றால் சம்மந்தப்பட்ட அனைவரும் அதைக் கண்டறிவதற்கு முன்பே தரவை குறியாகப் பயன்படுத்தப்படும் குறியீடாக பறிமாற்ற வேண்டும். எவரேனும் முக்கியத் தகவலை இடைமறித்தால் அவர்கள் எல்லா செய்திகளையும் படிக்கலாம்.

Chapter 17.indd 314 03/09/18 5:52 PM



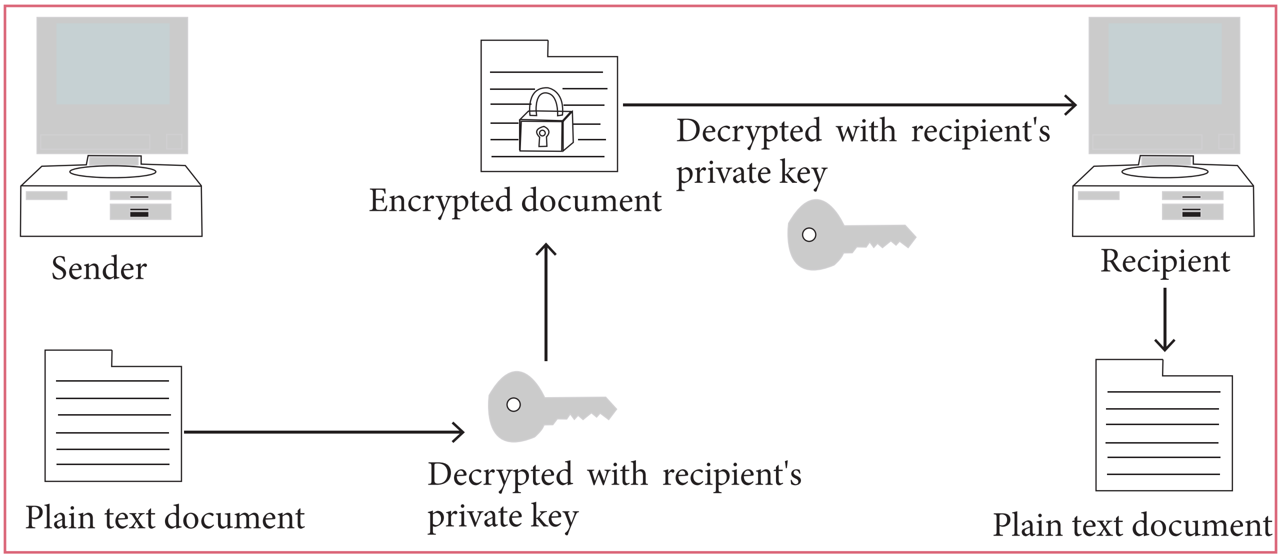
படம் 17.10 சமச்சீர் குறியீடு குறியாக்கத்தின் செயல்பாட்டை விவரிக்கிறது.

பொது குறியாக்கம்

பொது குறியீடு குறியாக்கத்தை சமச்சீரற்ற குறியாக்கவியல் என்று அழைக்கப்படுகின்றது. இது முக்கிய மதிப்பின் ஜோடியின் கருவியை பயன்படுத்துகிறது.

குறியாக்கம் மற்றும் மறை குறியாக்கம் செயல்முறைக்கு வேறுபட்ட குறியீடு பயன்படுகிறது. குறியீடு ஒன்று என்பது பொதுவாக தனிப்பட்ட குறியீடு என்று அழைக்கப்படுகிறது. மற்றொன்று பொது குறியீடு என்று அழைக்கப்படுகிறது.

தனிப்பட்ட குறியீடு உரிமையாளரால் இரகசியமாக வைக்கப்பட்டிருக்கிறது. பொதுகுறியீடு பகிர்ந்து கொள்ளப்படுகிறது அல்லது பொது மக்களுக்கு அதிகமாக கிடைக்கிறது. பெருநபரின் பொதுவிசையுடன் குறியாக்கப்பட்ட தரவு தொடர்புடைய தனிப்பட்ட குறியீடு மூலமே குறியாக்கம் செய்யப்படும்.



படம் 17.11 பொதுவிசை குறியாக்கத்தை காட்டுகிறது.

படிவூட்டல் சான்றிதழில் சமச்சீரற்ற குறியாக்கம்

படிவூட்டல் சான்றிதழ்களின் சமச்சீரற்ற குறியாக்கம் என்பது ஒரு இணையத்தின் சர்வர்-கிளைண்ட் தொடர்பு ஒரு உதாரணமாகும்.

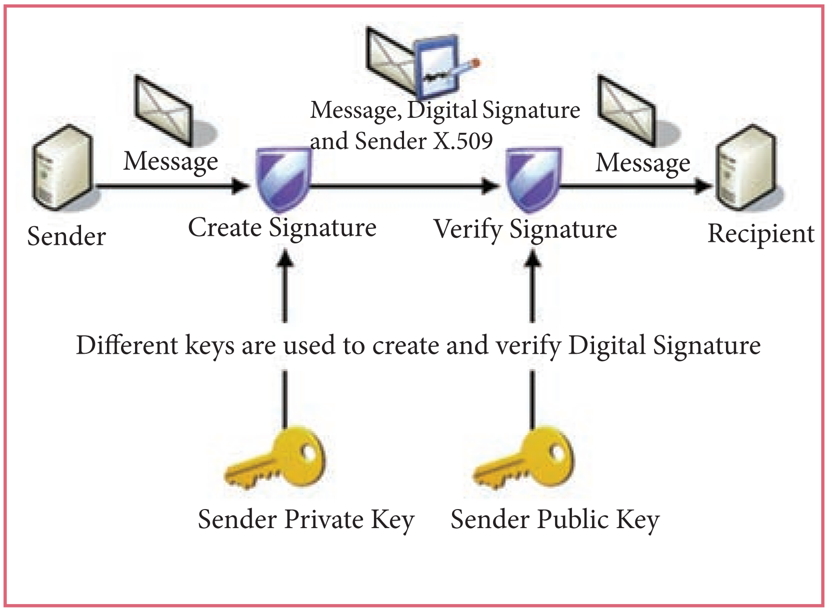
Chapter 17.indd 315 03/09/18 5:52 PM

ஒரு சான்றிதழ் என்பது ஒரு பயனர் மற்றும் சேவையகத்தை அடையாளம் காட்டும் தகவல் தொகுப்பு ஆகும். இது நிறுவனத்தின் பெயர், சான்றிதழை வழங்கிய நிறுவனம் பயனர்களின் மின்னஞ்சல் முகவரி, நாடு மற்றும் பயனரின் பொது குறியீடு போன்ற தகவல்களை கொண்டுள்ளது.

சேவையகம் மற்றும் இணையப் பாதுகாப்பான மறைகுறியாக்கம் செய்யப்பட்ட தொடர்பு, தேவைப்படும் போது குறியீட்டை பிறரிடம் அனுப்புவார்கள், இது சான்றிதழின் நகலை அனுப்புகிறது. ஒரு சான்றிதழை வைத்திருப்பவர் தனிப்பட்டவரின் அடையாளத்தை காண பயன்படுத்தலாம்.

டிஜிட்டல் கையொப்பம்

டிஜிட்டல் கையொப்பம் என்பது சமச்சீரற்ற குறியாக்கத்தின் அடிப்படையிலானது மின்னணு ஆவணத்தின் நிலை மின்னணு ஆவண பரிவர்த்தனை அல்லது செய்தியின் தோற்றம் அல்லது அடையாளம் மற்றும் நிலைக்கான ஆதாரங்களை வழங்க முடியும். அத்துடன் கையொப்பமிட்ட தகவல்களையும் ஒப்புகொள்ளுதல் ஆகும். படம் 17.12-ல் காட்டப்பட்டுள்ளது.



படம் 17.12 –டிஜிட்டல் கையொப்பம்

17.4 தகவல் தொழில்நுட்ப சட்டம் அறிமுகம் (Introduction to Information Technology Act)

21ஆம் நூற்றாண்டில், கணிப்பொறி, இணையதள மற்றும் ஐ.சி.டி அல்லது மின்னணு புரட்சி என்பது மனிதனின் வாழும் முறையை மாற்றி உள்ளது. இன்று காகிதத்தை அடிப்படையாகக் கொண்டு உள்ள தொடர்பு சாதனம், மின்னணு தொடர்பு சாதனமாக மாறி உள்ளது. அதன் படி நாம் பெற்றுள்ள புதிய கலைச் சொற்களான இணைய உலகம் மின்னணு பறிமாற்றம். மின்னணு வங்கி, மின்னணு வருமான வரி பிடித்தம் மற்றும் மின்னணு ஒப்பந்தம் முதலியன. இது ஒரு பக்கம் மின்னணு புரட்சி நன்மையாக இருந்தாலும், மற்றொரு பக்கம் தீமைகளும் உள்ளன. இணைய சேவை மற்றும் ஐ.சி.டி ஆகியவை குற்றவாளிகள் கைகளில், குற்றமிழைப்பதற்கு ஆயுதமாக உள்ளது.

ஆகவே, புதிய உறுப்பினர்களை கொண்ட குழு அமைக்கப்பட்டு இணைய உலகில் உள்ள இணைய குற்றங்கள் கையாளப்பட வேண்டும். ஆகவே, இணைய சட்டங்கள் அல்லது இணைய வெளிச் சட்டங்கள் அல்லது தகவல் தொழில்நுட்ப சட்டங்கள் அல்லது இணைய சட்டங்கள் கொண்டு கையாளப்பட வேண்டும்.

Chapter 17.indd 316 03/09/18 5:52 PM

உங்களுக்குத் தெரியுமா?

இணையதளத்தை பயன்படுத்தவில்லை என்றால் இணைப்பை துண்டிக்க வேண்டும்.

மின்னஞ்சல்களை உங்களுக்கு அறிமுகம் இல்லாதவர்களுக்கு அனுப்பக் கூடாது.

ஆன்டி நச்சுநிரலை நிறுவி மேம்பாடு செய்து (Update) கொள்ள வேண்டும்..

N2MF7Y

மதிப்பீடு

பகுதி அ

I சரியான விடை யை தேர்வு செய்யவும் :

1. கீழ் கண்டவனவற்றில் எது செயல்முறை, பயிற்சி மற்றும் மதிப்பு தொடர்புடையது?

   அ. உரிமையில்லா நகலாக்கம் ஆ. நிரல்கள்

   இ. நச்சு நிரல்கள் ஈ. கணிப்பொறி நன்னெறி

2. வணிக நிரல்களை பொது சட்ட விரோதமாக பயன்படுத்துவது

   அ. இலவச பொருள் ஆ. வேர்ஸ்

   இ. இலவச மென்பொருள் ஈ. மென்பொருள்

3. கீழ்கண்டவற்றுள் எது தானே பெருக்கிக் கொள்ளவும் மற்றும் இணைத்துக் கொள்ளவும் கணிப்பொறி நிரல்கள் தேவையில்லாதது?

   அ. நச்சுநிரல் ஆ. வார்ம்ஸ்

   இ. ஸ்லைவேர் ஈ. ட்ரோஜன்

4. கீழ்கண்டவற்றுள் எது பயனர் இணைய தளத்தை பார்வையிடுகிறது?

   அ. ஸ்பைவேர் ஆ. குக்கிகள்

   இ. வார்ம்ஸ் ஈ. ட்ரோஜன்

5. கீழ்கண்டவற்றில் எது தீங்கிழைக்கும் நிரல்கள்?

   அ. வார்ம்ஸ் ஆ. ட்ரோஜன்

   இ. ஸ்பைவேர் ஈ. குக்கிகள்

6. கணிப்பொறி வலைப்பின்னல் வழியாக உள்நுழையவும் வெளியேறும் சமிக்ஞைகளை கண்காணிக்கவும் கட்டுப்படுத்தவும் வகை செய்வது

   அ. குக்கிஸ் ஆ. நச்சுநிரல்

   இ. பயர்வால் ஈ. வார்ம்ஸ்

7. சிபர் எழுத்தை தனி எழுத்தாக மாற்றம் செய்யும் முறை

   அ. குறியாக்கம் ஆ. மறை குறியாக்கம்

   இ. நச்சு நிரல்கள் ஈ. பிராக்ஸி சேவையகம்

Chapter 17.indd 317 03/09/18 5:52 PM

8. இ- வணிகம் என்பது

அ. மின்னணு வணிகம் ஆ. மின்னணு தரவு மாற்றம்

இ. மின்சார தரவு மாற்றம் ஈ. மின்னணு வணிகமயமாக்க

9. சேவையற்ற மின்னஞ்சல் அடுத்தவர்களுக்கு பறிமாற்றம் செய்தல்

அ. ஊழல் ஆ. ஸ்பேம் – மின்னஞ்சல் குப்பைகள்

இ. மோசடி ஈ. ஸ்பூலிங் (சுருளாக்கம்)

10. பரிமாற்றத்திற்கான சட்ட அனுமதியை செயல்படுத்துவது

அ. மின்னணு தரவு உள் பறிமாற்றம் ஆ. மின்னணு தரவு பரிமாற்றம்

இ. மின்னணு தரவு மாற்றம் ஈ. மின்சார தரவு பரிமாற்றம்

பகுதி– ஆ

II குறுவினாக்கள் (2 மதிப்பெண்கள்)

1. ஹார்வஸ்டிங் என்றால் என்ன?
2. வார்ஸ் என்றால் என்ன?
3. விளசல் பற்றி சிறுகுறிப்பு.
4. இரண்டு வகையான இணையதள தாக்குதல் பற்றி எழுதுக.
5. குக்கி என்றால் என்ன?

பகுதி– இ

III சிறுவினாக்கள் (3 மதிப்பெண்கள்)

1. பையர்வாலின் பங்கு பற்றி எழுதுக?
2. குறியாக்கம் மற்றும் மறையாக்கம் பற்றி எழுதுக.
3. குறியாக்கம் சமச்சீர் குறியீடு பற்றி விளக்குக.
4. கணினி பயனர் பின்பற்றும் வழி காட்டுதல்கள் பற்றி எழுதுக?
5. நெறி முறை சிக்கல் என்றால் என்ன? பெயர்களை எழுதுக.

பகுதி – ஈ

IV பெருவினாக்கள் (5 மதிப்பெண்கள்)

1. கணிப்பொறி பயன்படுத்தும் போது ஏற்படும் பல்வேறு குற்றங்கள் யாவை?
2. களவாடல் என்றால் என்ன? களவாடலின் வகைகள் யாவை? மேலும் அதை எவ்வாறு தடுக்கலாம்?
3. இணையதள தாக்குதலின் வகைகள் யாவை?

துணை நூல்கள்:

- Computer Network Security and Cyber Ethics by Joseph MiggaKizza
- "Investigating Cyber Law and Cyber Ethics: Issues, Impacts and Practices: 1" by Alfreda Dudley and James Braman

கணிப்பொறியில் தமிழ்

அறிமுகம்

“ பிறநாட்டு நல்லறிஞர் சாத்திரங்கள்
தமிழ்மொழியிற் பெயர்த்தல் வேண்டும்;
இறவாத புகழுடைய புதுநூல்கள்
தமிழ்மொழியில் இயற்றல் வேண்டும்;
மறைவாக நமக்குள்ளே பழங்கதைகள்
சொல்வதிலோர் மகிமை இல்லை;
திறமான புலமையெனில் வெளிநாட்டோர்;
அதை வணக்கஞ் செய்தல் வேண்டும். ”

- ■ மகாகவி பாரதி

N2R2VV

இருபதாம் நூற்றாண்டின் இணையற்ற மனித கண்டுபிடிப்புதான் கணிப்பொறி. தொடக்க காலத்தில் கணக்கிடும் கருவியாக தொடங்கிய கணிப்பொறியின் வளர்ச்சி இன்று கணிப்பொறி இல்லாமல் மனித வாழ்வு இல்லை என்ற நிலையில் தனது அசுர வளர்ச்சியால் மனிதர்களின் அத்தியாவசிய பொருளாக மாறிவிட்டது.

மாறிவரும் தொழில்நுட்பங்களுக்கு ஈடுகொடுத்து தன்னை தகவமைத்துக்கொள்ளும் ஆற்றல் எந்த மொழிக்கு இல்லையோ அந்த மொழி கால வெள்ளத்தில் அடித்து செல்லப்பட்டுவிடும் என்பது உண்மை. எத்தனை புதிய தொழில்நுட்பங்கள் வந்தாலும் அவை அனைத்தையும் தன்வயப்படுத்திக்கொண்டு தன்னிலை மாறாது, வாழும் மொழியாக பல ஆயிரம் ஆண்டுகளாக வாழ்ந்து வருவது நமது செந்தமிழ். தமிழ் வெறும் மொழியல்ல, நமது அடையாளம், நமது உயிர், நமது உணர்வு.

“எங்கள் வாழ்வும், எங்கள் வளமும் மங்காத தமிழென்று சங்கே முழங்கு ” – புரட்சி கவி.

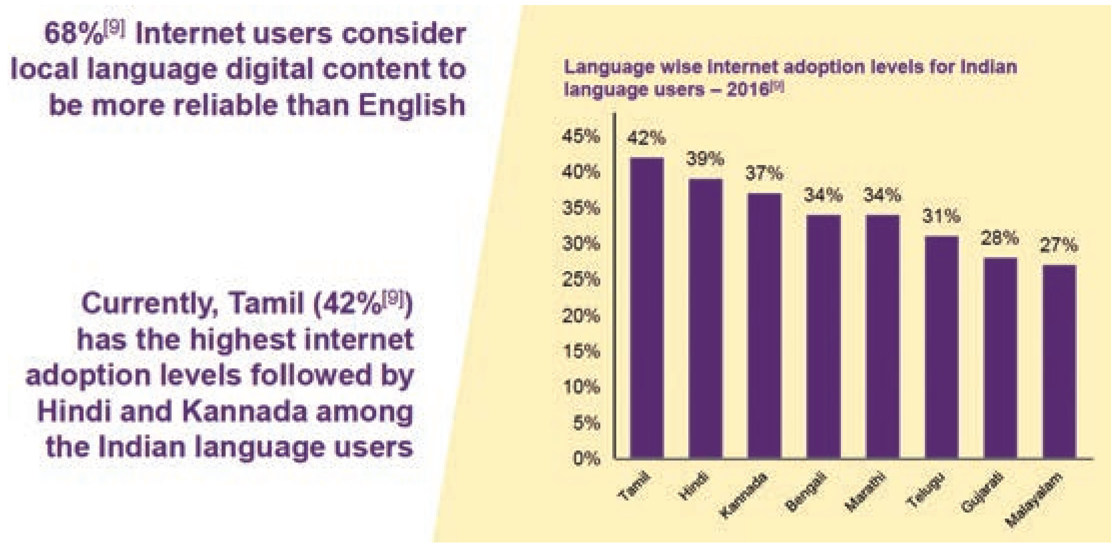
இயல், இசை, நாடகம் எனும் முத்தமிழ் இன்று கணினித்தமிழ் என “நான்தமிழ்” (நான்கு + தமிழ்) என்றாகி தமிழ்த்தாயின் முடியில் மேலும் ஓர் அணியாய் திகழ்கின்றது. கணிப்பொறியில் தமிழின் நிலை அதன் எதிர்காலம் பற்றி இனி காணாலாம்.

18.1 இணையத்தில் தமிழ்:

இணையம் இன்று ஒவ்வொரு சாதாரண மனிதனின் வாழ்விலும் பிரிக்கவியலா ஓர் அங்கமாகி விட்டதை நாம் அறிவோம். இணையம் மிகச்சிறந்த தகவல் தொழில்நுட்ப சாதனம். இணையத்தின் மூலம் நாம் அனைத்தையும் தெரிந்துகொள்ளலாம்.

சிங்கப்பூர்-யை தலைமையிடமாகக் கொண்ட KPMG என்ற நிறுவனம் கூகுளு-டன் இணைந்து கடந்த ஆகஸ்டு 2017-ல் நடத்திய ஆய்வில், 2016ம் ஆண்டுவாக்கில் இந்தியாவில் இணையத்தில் அதிகமாக பயன்படும் மொழிகளில் தமிழ் முதலிடம் வகிப்பதாகவும், 42% பேர் தமிழ் மூலம் இணையத்தை பயன்படுத்துவதாகவும் தெரிவித்துள்ளது.

Chapter 18.indd 319 03/09/18 5:52 PM

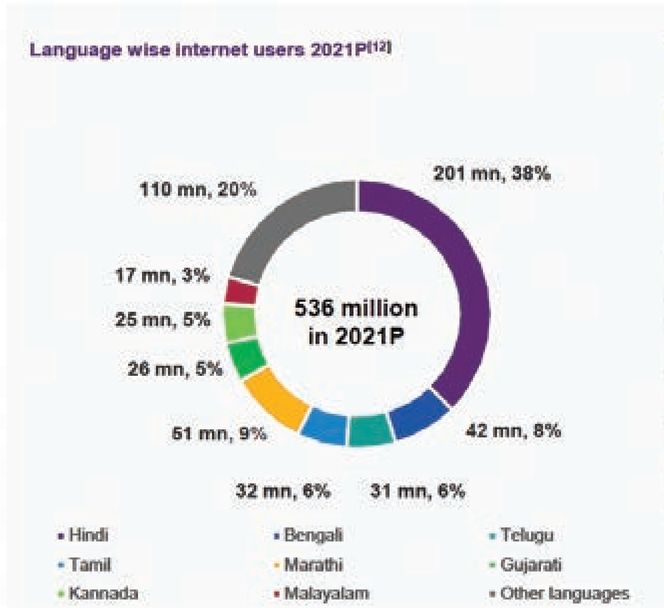


மேலும், 2021ம் ஆண்டுவாக்கில், 74% பேர் தமிழில் இணையத்தை அணுகுவார்கள் என்றும் அப்போதும், இந்தியாவில் இணையத்தில் தமிழே முதலிடத்தில் இருக்கும் எனவும் தெரிவிக்கப்பட்டுள்ளது.

By 2021, the number of Hindi internet users will be more than English users

Marathi, Bengali, Tamil and Telugu internet users are expected to form 30%[12] of the total Indian language internet user base

Tamil and Kannada speakers have the highest propensity to adopt internet in future with the Indian language enablement of the ecosystem



Language	Adoption propensity
Hindi	54%
Bengali	46%
Telugu	60%
Tamil	74%
Marathi	43%
Gujarati	43%
Kannada	74%
Malayalam	44%
Other languages	NA*

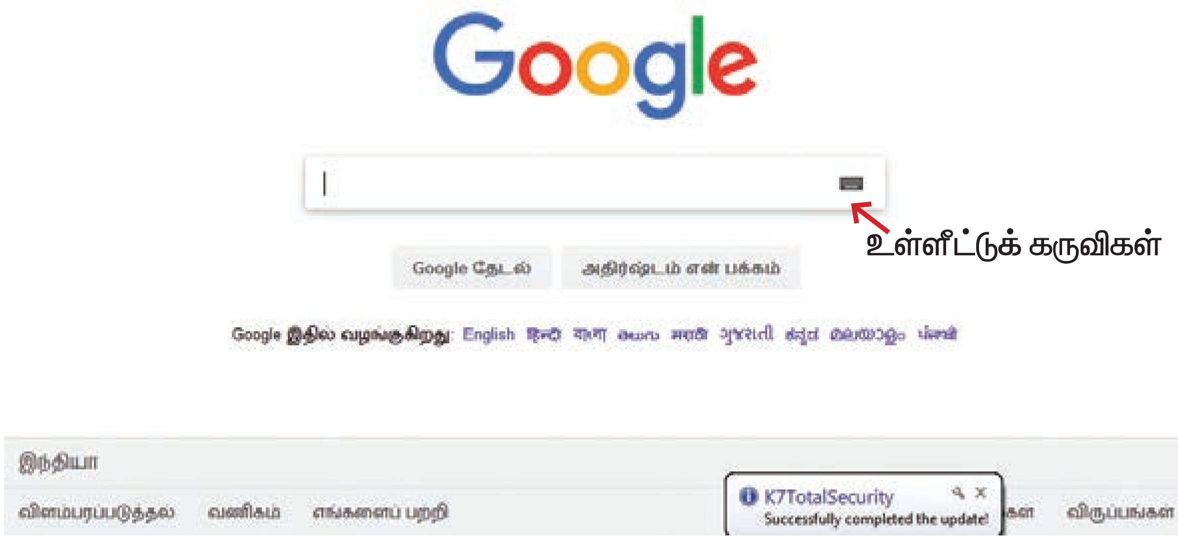
* Propensity to adopt internet in future has been ascertained for select 8 Indian languages

இந்த புள்ளிவிவர கணக்குகள் மேலும் பல இணைய சேவைகளை தமிழில் வழங்க உதவும்.

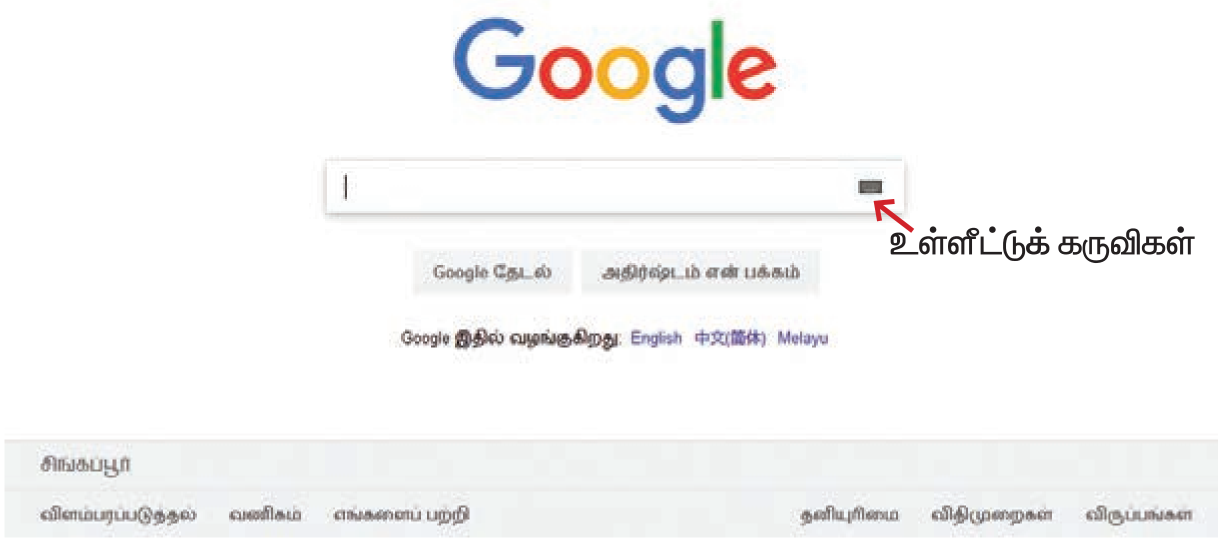
18.2 தமிழில் தேடுபொறிகள்:

இணையம் என்ற அகன்ற வெளியில், உங்களுக்கு தேவையான தகவல்களை நொடிப்பொழுதில் தேடித்தருபவை "தேடுபொறிகள்". பல தேடுபொறிகள் இருந்தாலும், அவற்றில் சிறந்து விளங்குபவை ஒரு சில மட்டுமே. இணைய உலகில் சிறந்த பத்து தேடுபொறிகளில், "கூகுள் (Google)" முதலிடத்திலும், மைக்ரோசாஃப்ட் நிறுவனத்தின் "பிங் (Bing)" இரண்டாம் இடத்திலும், "யாகூ (Yahoo)" மூன்றாமிடத்திலும் உள்ளது. இதில், கூகுள் மற்றும் பிங் தமிழில் தேடும் சேவைகளை வழங்கி வருகின்றது. கூகுள் தமிழில் தேடுவதற்கு வசதியாக, தமிழ் தட்டச்சு வசதியையும் வழங்குகின்றது.

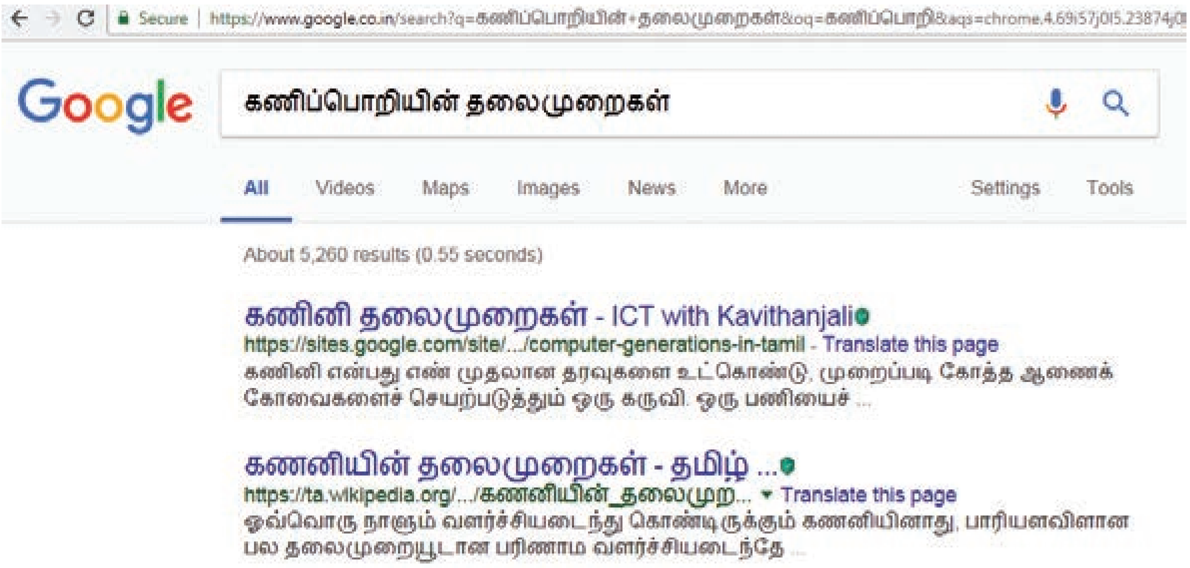
Chapter 18.indd 320 03/09/18 5:52 PM



படம் 18.1(a) தமிழில் கூகுள் (இந்தியா)



படம் 18.1(b) தமிழில் கூகுள் (சிங்கப்பூர் மற்றும் மலேசியா)

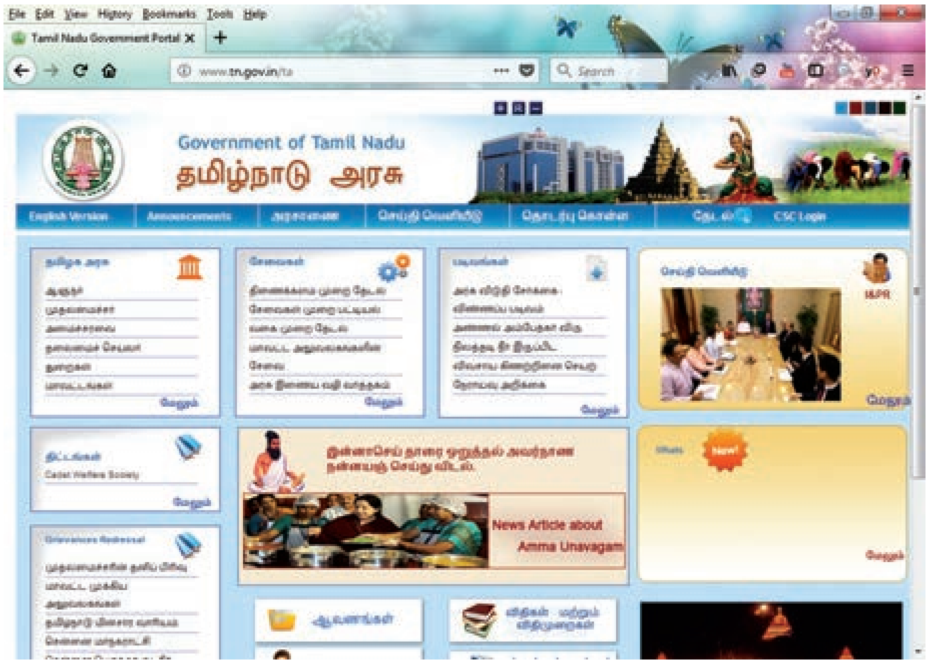


படம் 18.2 கூகுளில் தமிழில் தேடுதல்

Chapter 18.indd 321 03/09/18 5:52 PM

18.3 மின் அரசாண்மை (e-Governance):

ஒரு அரசின் சேவைகளை இணையத்தின் வழியே பெறுவது மின் அரசாண்மை எனப்படும். நமது தமிழ்நாடு அரசு தனது அனைத்து சேவைகளையும் இணையத்தின் வழியே வழங்கி வருகின்றது. தமிழகத்தின் எந்த மூலையில் இருந்தாலும், அரசுடன் உடனடியாக இணையத்தின் வழியே தொடர்பு கொள்ள முடியும். முக்கிய அறிவிப்புகள், அரசாணைகள், அரசின் திட்டங்கள், சேவைகள் போன்றவை இணையத்தின் வழியே கிடைக்கின்றது.



படம் 18.3 தமிழக அரசின் இணையதளம்

தமிழ் வழி மின் அரசாண்மை	இணைய முகவரி
தமிழ்நாடு அரசு அதிகாரப்பூர்வ இணையதளம்	http://www.tn.gov.in/ta
வேளாண்மை பொறியியல் துறை	http://www.aed.tn.gov.in/
தமிழ்நாடு அரசு சுற்று சூழல் துறை	http://www.environment.tn.nic.in/
அரசு தேர்வுகள் துறை	http://www.dge.tn.nic.in/
தமிழ்நாடு சுகாதாரத் துறை	http://www.tnhealth.org/
தமிழ்நாடு சிறு, குறு தொழில் முனைவோர் நலத்துறை	http://www.msmeonline.tn.gov.in/
ஊரக வளர்ச்சி மற்றும் ஊராட்சித் துறை	http://www.tnrd.gov.in/
பிற்படுத்தப்பட்டோர், மிகவும் பிறபடுத்தப்பட்டோர் மற்றும் சிறுபான்மையினர் நலத்துறை	http://www.bcmbcmw.tn.gov.in/
தமிழ்நாடு அரசு வனத்துறை	https://www.forests.tn.gov.in/
இந்து அறநிலையத் துறை	http://www.tnhrce.org/
தமிழ்நாடு அரசு பணியாளர் தேர்வாணையம் (TNPSC)	http://www.tnpsc.gov.in/tamilversion/index.html
இலங்கை அரசின் அதிகாரப்பூர்வ இணையதளம்	https://www.gov.lk/index.php

Chapter 18.indd 322 03/09/18 5:52 PM

இந்தியாவிற்கு வெளியே, இலங்கை அரசு தனது இணைய சேவை முழுவதும் தமிழில் வழங்குகிறது. சிங்கப்பூர் அரசு தனது அதிகாரப்பூர்வ இணைய தளத்தை தமிழில் இயக்கவில்லை எனினும், அனைத்து அரசு சேவைகளை, தமிழிலும் வழங்கி வருகின்றது

18.4 மின் நூலகம்:

மின்நூல்களை (e-Books) தொகுத்து வழங்குபவை மின் நூலகங்கள், லட்சக்கணக்கான தமிழ்நூல்களைக் கொண்ட மின்நூலகங்கள் பல நிறுவப்பட்டுள்ளது. இவை, இணையத்தின் வழியே அச்சு புத்தகங்கள் போய் சேரமுடியாத இடங்களில் வாழும் ஆயிரக்கணக்கான தமிழர்களுக்கு நல்ல பல தமிழ்நூல்களை கொண்டு சேர்க்கும் பணியினை செய்து வருகின்றன.

மின் நூல்கள் கிடைக்கும் தளங்கள்	இணைய முகவரிகள்
ஒன்றாம் வகுப்பு முதல் 12 ம் வகுப்பு வரையான தமிழ்நாடு மாநில பள்ளி பாடப்புத்தகங்களும், இடைநிலை ஆசிரியர் பயிற்சி பாடப்புத்தகங்களும்	http://www.textbooksonline.tn.nic.in/
தமிழ்நாடு இணைய கல்விக் கழக மின் நூலகம் (முன்னர் தமிழ்நாடு இணையப் பல்கலைக் கழகம் என அழைக்கப்பட்டது)	http://www.tamilvu.org/library/libindex.htm
கன்னிமரா மின் நூலகம்	http://connemarapubliclibrarychennai.com/Veettukku_oru_noolagam/index.html
தமிழிணைய மின்னூலகம்	http://tamildigitallibrary.in/
சென்னை நூலகம்	http://www.chennailibrary.com/
தமிழகம்	http://www.thamizhagam.net/parithi/parithi.html
மதுரை தமிழ் இலக்கிய மின் தொகுப்புத் திட்டம்	http://www.projectmadurai.org/pmworks.html
பழமையான புத்தகங்கள் மற்றும் கையெழுத்துப் பிரதிகள்	http://www.tamilheritage.org/old/text/ebook/ebook.html
நூலகம்	http://www.noolaham.org/wiki/index.php/
அண்ணா நூற்றாண்டு நூலகம்	http://www.annacentenarylibrary.org/

18.5 தமிழ் தட்டச்சு இடைமுக மென்பொருள்

கணிப்பொறியை பயன்படுத்தி சொற்செயலிகளில் ஆவணங்களை தட்டச்சு செய்யவும், இணையத்தின் வழியே தகவல்களை தேடவும், பெறவும் மட்டுமே தமிழ் அதிகமாக பயன்படுத்தப்படுகிறது. கணிப்பொறியில் தமிழில் தட்டச்சு செய்ய பல்வேறு வழிமுறைகள் பயன்படுத்தப்படுகிறது. அவற்றில், தமிழ் இடைமுக மென்பொருள்களை பயன்படுத்தி தட்டச்சு செய்யும் முறை தற்போது மிகவும் பிரபலமான ஒன்றாகும். கணிப்பொறிகள் மட்டுமின்றி ஸ்மார்ட் கைப்பேசிகளிலும், இம்முறையில் தட்டச்சு செய்வது மிகவும் எளிதான ஒன்றாகும்.

Chapter 18.indd 323 03/09/18 5:52 PM

18.5.1 பிரபலமான தமிழ் இடைமுக விசைப்பலகைகள்:

- NHM Writer, e-கலப்பை மற்றும் லிப்பிகார் – போன்றவை ஆங்கில ஒலியியல் முறையில், தமிழ் யுனிகோட், தமிழ் 99 போன்ற தட்டச்சு முறையில் வேலை செய்யும் தமிழ் விசைப் பலகை இடைமுக மென்பொருட்கள் ஆகும்.

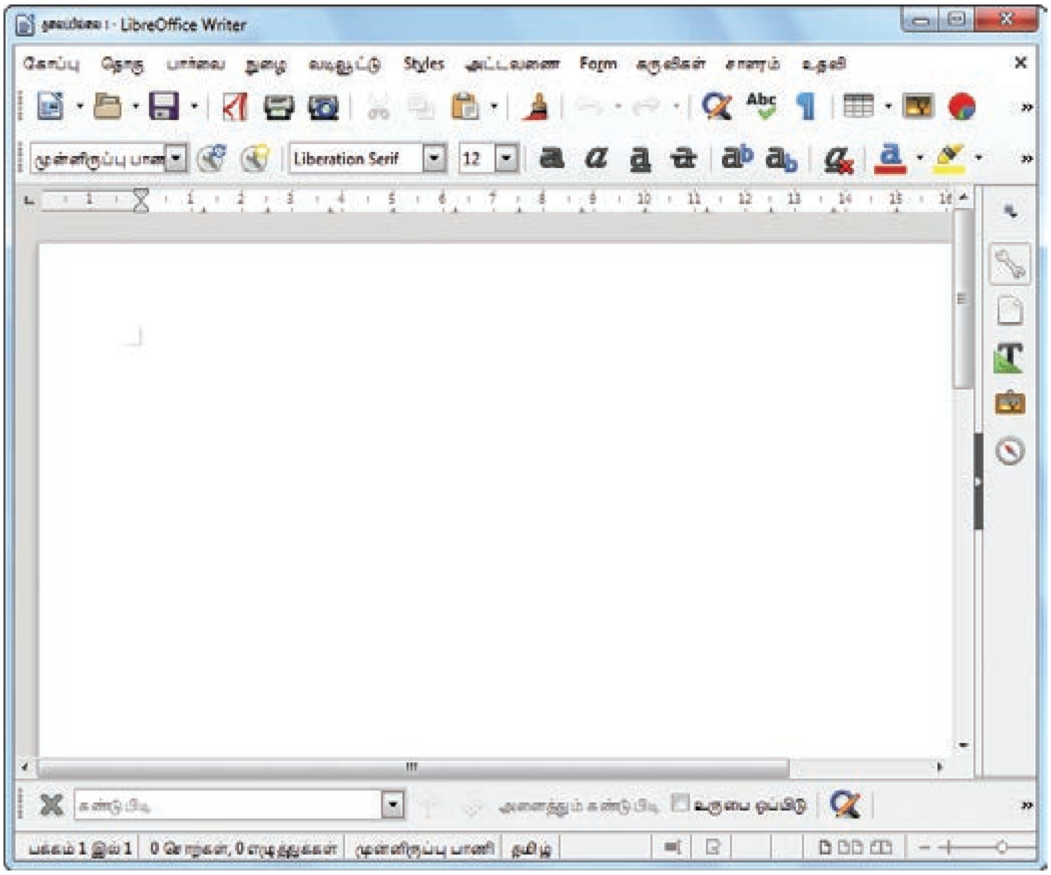
- செல்லினம் மற்றும் பொன்மடல் – ஸ்மார்ட் கைப்பேசிகளில், ஆண்ட்ராய்டு இயக்க அமைப்பில், ஆங்கில ஒலியியல் முறையில் பயன்படுத்தப்படும் இடைமுக விசைப்பலகை மென்பொருட்கள் ஆகும்.

படம் 18.4 eKalappai தொடக்க திரை

18.6 தமிழ் அலுவலக மென்பொருள்கள்

பிரபலமான அலுவலக மென்பொருள்களான மைக்ரோசாப்ட் ஆஃபீஸ், ஒபன் ஆஃபீஸ் போன்றவை தமிழில் பயன்படுத்தும் வகையில் தமிழ் இடைமுக வசதியை வழங்குகிறது. இவ்வசதிகளை இணையத்தின் வழியே பதிவிறக்கம் செய்து, நிறுவிக் கொள்ள முடியும். தமிழ் இடைமுகம் நிறுவப்பெற்ற பின்னர், அலுவலக மென்பொருட்களின் (மைக்ரோசாப்ட் ஆஃபீஸ், ஒபன் ஆஃபீஸ்) பட்டிப்பட்டை, குறும்படங்களின் பெயர்கள் போன்ற அனைத்தும் தமிழில் தோன்றும். மேலும், கோப்புகள் மற்றும் கோப்புரைகளின் பெயர்களை தமிழிலேயே உருவாக்க முடியும்.

Chapter 18.indd 324 03/09/18 5:52 PM



படம் 18.5 Libra Office Writer தமிழ் பதிப்பு

இவை மட்டுமின்றி, முழுவதும் தமிழிலேயே இயங்கும் அலுவலக மென்பொருள்களும் பயன்பாட்டில் உள்ளது. அவற்றில் தமிழ் லிபரெ ஆஃபீஸ், தமிழ் ஒபன் ஆஃபீஸ், அழகி யுனிகோட் எடிட்டர், பொன்மொழி, மென்தமிழ், கம்பன், வாணி போன்ற அலுவலக மென்பொருள்கள் முழுவதும் தமிழில் பயன்படுத்தும் வகையில் வடிவமைக்கப்பட்டுள்ளது.

18.7 தமிழ் மொழிபெயர்ப்பு பயன்பாடுகள்:

"தமிழ்பொறி" என்ற மென்பொருள், சுமார் 30000 க்கும் அதிகமான ஆங்கில சொற்களுக்கு ஈடான தமிழ் சொற்களை உள்ளடக்கிது. இதன் மூலம், சிறிய ஆங்கில சொற்றொடர்களுக்கு இணையான தமிழ் மொழி பெயர்ப்பை பெறலாம்.

மேலும், கூகுள் நிறுவனம், தமிழிலிருந்து பிற மொழிகளுக்கு மொழி பெயர்க்கும் "மொழிபெயர்ப்பு" நிகழ்நேர பயன்பாட்டை வழங்கிவருகின்றது.

18.8 தமிழ் நிரலாக்க மொழி:

கணிப்பொறி மற்றும் ஸ்மார்ட் கைப்பேசிகளில் மென்பொருள்களை வடிவமைக்க பயன்படும் நிரலாக்க மொழிகள் இதுவரை ஆங்கிலத்தில் மட்டுமே இருந்துவந்த நிலையில், தமிழிலும், நிரலாக்க மொழி வடிவமைக்கும் முயற்சிகள் நடைபெற்றுவருகின்றன. அதனடிப்படையில், பைத்தான் நிரலாக்க மொழியை அடிப்படையாக கொண்டு, முதல் தமிழ் நிரலாக்க மொழி "எழில்" வடிவமைக்கப்பட்டுள்ளது. இந்த நிரலாக்க மொழியை பயன்படுத்தி, சிறிய அளவிலான நிரல்களை தமிழிலேயே எழுதமுடியும்.

18.9 தகவல் பரிமாற்றத்திற்கான தமிழ் குறியீட்டு முறை

TSCII (Tamil Script Code for Information Interchange)

கணிப்பொறி இருநிலை எண்களை அடிப்படையாக கொண்டது எனவும், கணிப்பொறியில் உள்ளிடப்படும் தரவுகள் இருநிலை எண்களாக மாற்றப்படும் என்பதையும், இந்த பாடப் புத்தகத்தின் முதல் அலகில் கற்றீர்கள். கணிப்பொறியில் கொடுக்கப்படும் தரவுகளும், தகவல்களும் கையாள **ASCII** என்ற குறியீட்டு முறை பயன்படுத்தப்பட்டுவந்தது. இந்தக் குறியீட்டு முறை ஆங்கில மொழியை மட்டுமே கையாளும் திறன் பெற்றது. எனவே, ஆங்கிலம் அல்லாத பிற மொழிகளை கணிப்பொறி உள்ளிட்ட மின்னணு சாதனங்களில் எளிதில் கையாள பொருத்தமான ஒரு குறியீட்டு முறை பற்றிய ஆய்வில், நமது தமிழ் மொழியை கையாள உருவாக்கப்பட்ட முதல் குறியீட்டு முறை தான் **TSCII (Tamil Script Code for Information Interchange)** என்பதாகும். இந்த குறியீட்டு முறை **2007**ம் ஆண்டில், **ICANN**-ன் **IANA (Internet Assigned Numbers Authority)**யில் பதிவு செய்யப்பட்டது. இந்த குறியீட்டு முறை தழுவியே தமிழுக்கான **Unicode** முறையும் வடிவமைக்கப்பட்டது.

தகவல் பரிமாற்றத்திற்கான இந்திய குறியீட்டு முறை

ISCII (Indian Script Code for Information Interchange)

இந்திய மொழிகளை கையாளுவதற்காக வடிவமைக்கப்பட்ட ஒரு குறியீட்டு முறை. இந்த முறை, ஒருங்குறி முறையுடன் இணைக்கப்பட்டுவிட்டது.

ஒருங்குறி Unicode:

தமிழ் உள்ளிட்ட பல்வேறு உலக மொழிகளை கையாளுவதற்காக வடிவமைக்கப்பட்ட ஒரு குறியீட்டு முறை தான் ஒருங்குறி என்று அழைக்கப்படும் யுனிகோட். இதன் முதல் பதிப்பான யுனிகோட் 1.0.0 என்பது அக்டோபர் 1991ல் அறிமுகப்படுத்தப்பட்டது. அறிமுகப்படுத்தப்பட்ட போது, தமிழ் உள்ளிட்ட 23 மொழிகளை கையாளும் திறன் பெற்றிருந்தது. தமிழை கையாள பல குறியீட்டு முறைகள் இருப்பினும், இன்று ஒருங்குறியே சிறந்த ஒன்றாக திகழ்கிறது.

18.10 தமிழ் இயக்க அமைப்புகள்

கணிப்பொறி அல்லது ஸ்மார்ட் கைப்பேசி போன்ற மின்னணு சாதனங்களை இயக்க ஏதேனும் ஒரு இயக்க அமைப்பு வேண்டும். தனியாள் கணிப்பொறிகளில் மிகப்பிரலமான இயக்க அமைப்பு மைக்ரோசாப்ட் நிறுவனத்தின், விண்டோஸ் ஆகும். விண்டோஸ் இயக்க அமைப்புக்கு அடுத்த இடத்தில், திறந்த மூல இயக்க அமைப்பான லினக்ஸ் உள்ளது.

ஒரு கணிப்பொறியை எளிதில் இயக்க இயக்க அமைப்பு பயன்படுகிறது. ஒரு இயக்க அமைப்பு பயன்படுத்துவதற்கு எளிதாகவும், பயன்படுத்துவோர் புரிந்துகொள்ளும் வகையிலும் அமைக்கப்பட்டிருக்க வேண்டும் என்ற வகையில், கணிப்பொறியில் பயன்படுத்தப்படும் விண்டோஸ், லினக்ஸ் போன்ற இயக்க அமைப்புகளையும், ஸ்மார்ட் போன்களில் பரவலாக பயன்படுத்தப்படும் ஆண்டிராய்டு இயக்க அமைப்பும் தமிழ் மொழியில் வழங்கப்பட்டுள்ளது.

விண்டோஸ் இயக்க அமைப்பின் பணிப்பட்டை, திரை முகப்பு, பணிக்குறிகள், கட்டளைகள் அனைத்தும் தமிழிலேயே காணக் கிடைக்கும். விண்டோஸ்-ன் தமிழ் இடைமுக வசதி, மைக்ரோசாப்ட் நிறுவனத்தின் இணையதளத்திலிருந்து பதிவிறக்கம் செய்து, நிறுவிக்கொள்ளலாம்.

Chapter 18.indd 326 03/09/18 5:52 PM

18.11 தமிழ் வளர்க்கும் நிறுவனங்களும், திட்டங்களும்:

தமிழ் இணையக் கல்விக்கழகம்:

தமிழை உலகெங்கிலும் இணையத்தின் வழியே கொண்டு சேர்க்கும் நோக்குடன், 2001ம் ஆண்டு பிப்ரவரி மாதம் 17ம் நாளில், தமிழ் இணையப் பல்கலைக் கழகம், தமிழக அரசால் தொடங்கப்பட்டது. தற்போது தமிழ் இணையக் கல்விக் கழகம் என்ற பெரியில் செயல்பட்டு வரும் இந்நிறுவனம் இணையத்தின் வழியை தமிழ் மொழி, தமிழர் கலாச்சாரம், பண்பாடு போன்றவை உலகெங்கிலும் வாழும் மக்களுக்கு வழங்குவதற்காக, மழலையர் கல்வி முதல் பட்டப் படிப்பு வரை பல்வேறு கல்வித் திட்டங்களை செயல்படுத்தி வருகின்றது.

இணைய முகவரி: **http://www.tamilvu.org/index.php**

வளர் தமிழ் இயக்கம், சிங்கப்பூர்

சிங்கப்பூரில் தமிழ்மொழியை வாழும் மொழியாகவும் பயன்பாட்டு மொழியாகவும் நிலைபெறச் செய்வதோடு, அனைத்து வயதினரையும் தமிழால் ஒன்றிணைத்து, சிங்கப்பூரில் தமிழ்மொழியை ஓங்கச் செய்யும் நோக்கத்துடன் 2001ம் ஆண்டில், சிங்கப்பூர் தகவல் கலை அமைச்சகத்தின் மேற்பார்வையில் இந்த இயக்கம் தொடங்கப்பட்டது.

இணைய முகவரி: **http://tamil.org.sg/ta**

மதுரை தமிழ் இலக்கிய மின் தொகுப்புத் திட்டம்:

தமிழில் உள்ள அனைத்து நூல்களையும், இணையத்தில் மின்னணு வடிவில் மாற்றம் செய்வதற்கான ஒரு தன்னார்வ அரசு சாரா திட்டம். இத்திட்டத்தின் மூலம் 1998ம் ஆண்டு முதல், இது வரை தமிழில் வெளிவந்துள்ள அனைத்து நூல்களையும் இணையத்தில் ஏற்றும் பணிகள் மேற்கொள்ளப்படுகிறது. உலகெங்கிலும் வாழும் தமிழர்களின் தன்னலமற்ற முயற்சியால் இத்திட்டத்தில் இதுவரை 600 நூல்கள் இணைய வெளியில் பதிவேற்றம் செய்யப்பட்டுள்ளது.

இணைய முகவரி: **http://www.projectmadurai.org/**

தமிழ் விக்கிபீடியா:

விக்கிபீடியா ஒரு திறந்த நிலை களஞ்சியமாகும். விக்கிபீடியாவில் எவரும், எந்த தலைப்பிலும் கட்டுரைகளை எழுதலாம். தமிழ் விக்கிபீடியா தமிழில் ஒரு இலட்சத்திற்கும் மேலான கட்டுரைகளை உள்ளடக்கிய ஒரு மிகப் பெரிய கலைக் களஞ்சியமாகும்.

இணைய முகவரி: **Web Site: https://ta.wikipedia.org/**

தமிழை, வாழும் மொழியாக்க வேண்டுமெனில், தொழில்நுட்பத்தில் வளர்ச்சியில், தமிழை பங்குபெற செய்ய வேண்டியது தமிழராகிய நமது கடமை. தனது இனத்தின் விழுமியங்களை மறந்தவர், தொலைந்து போனவர்கள். நாம் எத்தனை பெரிய தொழிநுட்பங்களை கற்றாலும், அதனுள் நமது இனத்தின் அடையாளமான தமிழை கொண்டு சேர்க்க வேண்டும். உலகின் முதல் மொழி, மூத்த மொழி ஐந்தாயிரம் ஆண்டுகளுக்கும் மேலாக இன்றைக்கும் வாழும் மொழியை, கணினி தொழில்நுட்பத்துடன் இணைப்பது நமது கடமை.

நினைவில் கொள்க

- இந்தியாவில் 2016 ஆம் ஆண்டு இறுதியுடன், தமிழ் மொழி முதன்மையான பட்டியலுடன் விளங்கி, இணையம் 42 விழுக்காடு பயன்படுத்துகின்றனர்.
- தமிழில் தேடுதல் வசதிகளை கூகுள் (Google) மற்றும் பீங் (Bing) அளிக்கின்றன.
- இணையத்தின் மூலமாக அரசு சேவைகளைப் பெற இ-சேவை வழங்குகின்றன.
- ஆயிரக்கணக்கான தமிழ் புத்தகங்கள் மின்னணு தமிழ் சேவைகள் மின்னனு புத்தக வடிவமாகி இலவசமாக கிடைக்கின்றன.
- ஆங்கில எழுத்துகளுக்கு இணையாக தமிழ் எழுத்துக்கள் 30,000 மொழிபெயர்ப்பு செயலி தமிழ் பொறியில் இருக்கிறது.
- தமிழ் மொழி நிரலில் முதன்மையானது "எழில்"
- உலக மொழிகளை கையாள, தமிழையும் சேர்த்து ஒருங்குறி (Unicode) ஒன்றிணைத்து செய்கின்றது.
- ஒன்றிணைத்து திரட்ட பலவற்றை, தமிழில் கையாளத் தகுந்த ஒன்று ஒருங்குறி(Unicode)
- சன்னல் (Windows) தமிழ் சுற்றுச்சுழல் இண்டர்பேஸ் செயல் இறக்கி இணையத்தில் பின் புகுத்தலாம்.

மதிப்பீடு

N2SNKP

பின்வரும் வினாக்களுக்கு விடையளி

1. தமிழில் சேவைகளை வழங்கி வரும் தேடுபொறிகளை பட்டியலிடுக
2. ஆண்ட்ராய்டு பயன்பாடு விசைப்பலகை என்றால் என்ன?
3. தமிழ் மென்பொருள் பயன்பாட்டு மொழி-சிறு குறிப்பு வரைக.
4. TSCII என்றால் என்ன?
5. தமிழ் வெர்சியுவல் அகாடமி சிறு குறிப்பு வரைக.

****

Chapter 18.indd 328 03/09/18 5:52 PM

சொற்களஞ்சியம்

N2HB9L

சொற்கள்	பொருள்
பாதிப்பு	தாக்கப்படுவது அல்லது பாதிக்கப்பட வாய்ப்புகள் உள்ளன.
நன்னெறி	நன்னெறி ஒரு நபரின் நடத்தையை நிர்வகிக்கும், தார்மீக கோட்பாடு ஆகும்.
இணையம்	கணிப்பொறி கலாச்சாரத்தின் தன்மை, தகவல் தொழில்நுட்பம் மற்றும் மெய் நிகர் உண்மை.
கணிப்பொறி குற்றம்	கணிப்பொறி குற்றம் என்பது புத்திசாலித்தனமான குற்றம்.
நம்பகத்தன்மை	உண்மையான அல்லது நம்பகமான தரம்.
சபோடேஜின்	திட்டமிட்டு அழிக்கவும் சேதப்படுத்தவும் அல்லது தடுக்கவோ இயலும்
குற்றம் புரிபவர்	தீங்கிழைக்கும் சட்ட விரோதமான அல்லது ஒழுக்க கேடான செயலை செய்வர்
உரிமையில்லா நகலாக்கம்	ஒரு நபர் சட்ட விரோதமான பதிப்புரிமை மீறுதல்.
ஹேக்கிங்	உரிமையாளர் அனுமதி இல்லாமல் கணினி மூலம் அங்கீகரிக்கப்படாத அணுகல்
கிராக்கிங்	சட்டபூர்வமாக பயனர்களுக்கு சேவையை மறுத்து நகலெடுக்கப்பட்ட பாதுகாப்பான திட்டத்தை உருவாக்க குறியீட்டை திருடுவது.
தீங்கிழைப்பது	வேண்டுமென்றே தீங்கு செய்வது.
இலவச பகிர்வு மென்பொருள்	இலவசமாக கிடைக்கும் ஒரு மென்பொருள்.
பகிர்வு மென்பொருள்	ஒரு குறிப்பிட்ட காலத்திற்கு ஒரு சோதனை அடிப்படையில் இலவசமாக விநியோகிக்கப்பட்ட ஒரு மென்பொருள்
ஃபிஷிங்	தீங்கிழைக்கும் தனிநபர் அல்லது தனி நபர்களின் குழுவொன்று மின்னஞ்சலை அனுப்புவதன் மூலம் அல்லது தனி நபரின் ஆன்லைன் வங்கி, கிரெடிட் கார்டு அல்லது பிற உள் நுழைவு தகவலை சேகரிக்க வடிவமைக்கப்பட்ட வலைதளங்களை உருவாக்குவதற்கு பயன்படுகிறது.
குக்கிகள்	இணைய தளங்களை பார்வையிடும் போது வலை சேவையகங்கள் உங்கள் வலை உலவிக்கு அனுப்பும் செய்தி
சேதப்படுத்துதல்	சட்ட விரோதமான தலையீடு சேதத்தை ஏற்படுத்தும் வகையில் குறுக்கீடு.
நச்சு எதிர்ப்பு	ஒரு குறிப்பிட்ட நச்சு நிரலுக்கு எதிர்ப்பு.
மோசடி	பெயரிடப்படாத அல்லது அடையாளம் தெரியாத மோசடி செய்தல் நேர்மையில்லாமல் ஏமாற்றுதல் சட்ட விரோதமாக நடத்தல்.
ஃபயர்வால்	ஒரு இணையப் பாதுகாப்பு ஆகும். இது முன்னரே தீர்மானிக்கப்பட்ட பாதுகாப்பு விதிகளின் அடிப்படையில் உள்வரும், வெளிச்செல்லும் சமிக்ஞைகளை கண்காணித்து கட்டுப்படுத்தும்.
பதிலாள் சேவையகம்	பதிலி சேவையகம் (Proxy Server) என்பது, உள்நிலை வலையமைப்பு மற்றும் இணையம் போன்ற பெரிய அளவிலான வலையமைப்புகளுக்கு நுழைவாயில் ஆகும். புராக்ஸி சேவையகம் அதிக செயல்திறன் மற்றும் பாதுகாப்பு வழங்குகிறது.
குறியாக்கம்	ஒரு செய்தியை அல்லது தகவலை குறியாக்க செயல்முறை அங்கீகரிக்கப்பட்ட பயனர்கள் அதை கையாளலாம்.
மறை குறியாக்கம்	இது சாதாரண உரைக்கு மீண்டும் மாற்றுவதன் மூலம் மறை குறியாக்கப்பட்ட உரை நீக்கப்படும் செயல்முறை.

கணினி அறிவியல்
மேல்நிலை முதலாமாண்டு – தொகுதி–2
வல்லுநர்கள், மேலாய்வாளர்கள் மற்றும் நூலாசிரியர்கள் பெயர் பட்டியல்

பாடப்பொருள் வல்லுநர்கள்

முனைவர் **T. V. கோபால்,**
பேராசிரியர்,
கணினி அறிவியல் மற்றும் தொழில்நுட்பத் துறை, பொறியியல் கல்லூரி,
அண்ணா பல்கலைகழகம், கிண்டி, சென்னை.

முனைவர் **பெ. இராதா**
உதவி பேராசிரியர், தகவல் தொழில்நுட்பவியல் துறை,
அரசினர் கலை மற்றும் அறிவியல் கல்லூரி, கோயம்புத்தூர்.

முனைவர் **நெஸ்டர் ஜெயக்குமார்**
இணைப் பேராசிரியர் மற்றும் துறைத்தலைவர்,
கணிப்பொறியியல் துறை, இலயோலா கல்லூரி, சென்னை.

க. சங்கர்
உதவி பேராசிரியர், கணினி அறிவியல் துறை,
RKM விவேகானந்தா கல்லூரி, மைலாப்பூர், சென்னை.

மேலாய்வாளர்கள்

முனைவர் **ரஞ்சனி பார்த்தசாரதி**
பேராசிரியர், தகவல் மற்றும் அறிவியல் தொழில்நுட்பம்,
பொறியியல் கல்லூரி, அண்ணா பல்கலைகழகம், கிண்டி, சென்னை.

E. முனிவேல்
விஞ்ஞானி/பொறியாளர் 'C'
தகவல் தொழில்நுட்பம் (தகவல் பாதுகாப்பு),
NIELI, NIT வளாகம், காலிகட், கேரளா.

தமிழ் வாசிப்பாளர்

ம. பாலகுருசாமி
தமிழ் மற்றும் கல்வியியல் பேராசிரியர்,

முகமது சதக்
ஆசிரியர் பயிற்சி கல்லூரி,
இராஜீவ் காந்தி சாலை, படூர், காஞ்சிபுரம் மாவட்டம்.

பாடநூல் ஒருங்கிணைப்பாளர்கள்

திரு. இரவிகுமார் ஆறுமுகம்,
துணை இயக்குநர்,
மாநிலக் கல்வியியல் ஆராய்ச்சி மற்றும் பயிற்சி நிறுவனம், சென்னை.

திருமதி R. தமிழ்ச்செல்வி
பட்டதாரி ஆசிரியை, அரசு உயர் நிலைப் பள்ளி, பூனாம்பாளையம்,
திருச்சி மாவட்டம்.

நூலாசிரியர்கள்

க. கண்ணன்
முதுகலைப் பட்டதாரி ஆசிரியர்,
சென்னை பெண்கள் மேல்நிலைப் பள்ளி,
இராட்லர் தெரு, சென்னை.

வி.கோ. இராமகிருஷ்ணன்
முதுகலைப் பட்டதாரி ஆசிரியர்,
கர்நாடக சங்கா மேல்நிலைப் பள்ளி,
தி. நகர், சென்னை.

பிந்து மோகன்தாஸ்
முதுகலைப் பட்டதாரி ஆசிரியர்,
விஜயந்தா மாதிரி மேல்நிலைப் பள்ளி,
எச்.வி.எஃப் எஸ்டேட், ஆவடி, சென்னை.

நு .வீ. கௌரி சங்கர்
முதுகலைப் பட்டதாரி ஆசிரியர்,
சென்னை பெண்கள் மேல்நிலைப் பள்ளி,
நுங்கம்பாக்கம், சென்னை.

இரா. சீனிவாசன்
முதுகலைப் பட்டதாரி ஆசிரியர்
சாந்தோம் மேல்நிலைப் பள்ளி, சென்னை.

கோ. லெனின்
முதுகலைப் பட்டதாரி ஆசிரியர்,
சென்னை பெண்கள் மேல்நிலைப் பள்ளி,
சைதாப்பேட்டை, சென்னை.

ஆ. சங்கீதா
முதுகலைப் பட்டதாரி ஆசிரியர்,
அரசினர் மேல்நிலைப் பள்ளி,
இராசந்தாங்கல், திருவண்ணாமலை மாவட்டம்.

முனைவர் க.எ. வளர்மதி
முதுகலைப் பட்டதாரி ஆசிரியர்,
வேலம்மாள் வித்யாஷ்ரம் பள்ளி, சூரப்பட்டு, சென்னை.

இரா. கஜலஷ்மி
முதுகலைப் பட்டதாரி ஆசிரியர்,
ஜெயகோபால் கரோடியா இந்து வித்யாலயா மேல்நிலைப் பள்ளி,
மேற்கு மாம்பலம், சென்னை.

முனைவர் ஹ. வித்யா
முதுகலைப் பட்டதாரி ஆசிரியர்,
டி.ஏ.வி. ஆண்கள் மேல்நிலைப் பள்ளி, கோபாலபுரம், சென்னை.

தமிழ் மொழிபெயர்ப்பாளர்கள்

S. உஷா ராணி
முதுகலைப் பட்டதாரி ஆசிரியர்,
சென்னை ஆண்கள் மேல்நிலைப் பள்ளி
பந்தர் கார்டன் தெரு, பெரம்பூர், சென்னை.

S. சிவசரவணன்
முதுகலைப் பட்டதாரி ஆசிரியர்,
சென்னை மேல்நிலைப் பள்ளி, எம்.ஜி.ஆர். நகர், சென்னை.

கலை மற்றும் வடிவமைப்புக் குழு

வடிவமைப்பு
THY டிசைனர்ஸ் & கம்யூட்டர்ஸ், சென்னை..

அட்டை வடிவமைப்பு
கதிர் ஆறுமுகம்

In-House - QC
மனோகர் இராதாகிருஷ்ணன்
ஜெரால்ட் வில்சன்
ப. அருண் காமராஜ்

ஒருங்கிணைப்பு
ரமேஷ் முனிசாமி

தட்டச்சு
த. மீனா, SCERT.

விரைவுக் குறியீடு மேலாண்மைக் குழு

இரா. ஜெகநாதன்
இடைநிலை ஆசிரியர்
ஊ.ஒ.ந.நி. பள்ளி, கணேசபுரம்–போளூர், திருவண்ணாமலை மாவட்டம்.

ந. ஜெகன்
பட்டதாரி ஆசிரியர்,
அ.ஆ.மே.நி. பள்ளி, உத்திரமேரூர், காஞ்சிபுரம் மாவட்டம்.

ஜே.எப். பால் எட்வின் ராய்
பட்டதாரி ஆசிரியர்,
ஊ.ஒ.ந.நி. பள்ளி, இராக்கிப்பட்டி, சேலம் மாவட்டம்.

இந்நூல் 80 ஜி.எஸ்.எம். எலிகண்ட் மேப்லித்தோ தாளில் அச்சிடப்பட்டுள்ளது.
ஆப்செட் முறையில் அச்சிட்டோர்: