அலகுIII | C++ ஓர் அறிமுகம்

பாடம் 9

HZPL7I

C++ ஓர் அறிமுகம்

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. C++ மொழி செயல்முறை மற்றும் பொருள் நோக்கு நிரலாக்க முறைகள் இரண்டையும் ஆதரிக்கிறது. எனவே, C++ ஒரு **கலப்பின மொழி** ஆகும். C++ அதன் முன்னோடி 'சி' மொழியின் நீட்டிப்பு ஆகும். ஜேர்ன் ஸ்ட்ரெளஸ்ட்ரப் தனது புதிய மொழியை "இனக்குழுக்களுடன் சி" என்று பெயரிட்டார். C++ என்னும் பெயரை (சி பிளஸ் பிளஸ் என உச்சரிக்க வேண்டும்) ரிக் மாஸ்சிட்டி (Rick Mascitti) என்பவர் சூட்டினார்.

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 ஸ்ட்ரெளஸ்ட்ரப்(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.

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.

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.

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)

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 மட்டுமே அனுமதிக்கப்படுகின்றன. பிற சிறப்பு எழுத்துருக்கள் பெயரின் ஒரு பகுதியாக அனுமதிக்கப்படாது.

Chapter 9.indd 120 4/11/2020 11:44:40 AM

- 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.

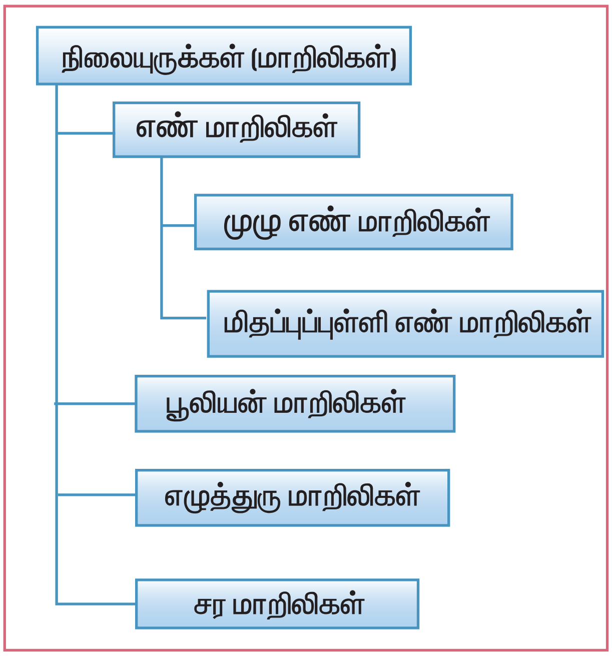
குறிப்பெயர்கள்	சரியா/ தவறா	தவறுக்கான காரணம்
Num	சரி	
NUM	சரி	
_add	சரி	
total_sales	சரி	
tamilMark	சரி	
num-add	தவறு	சிறப்பு எழுத்துரு (-) கொண்டுள்ளது
this	தவறு	இது ஒரு சிறப்புச் சொல்லை கொண்டுள்ளது. சிறப்புச் சொல்லை குறிப்பெயராக பயன்படுத்த முடியாது.
2myfile	தவறு	குறிப்பெயர் எப்பொழுதும் எழுத்து அல்லது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

- மாறியின் பெயரை பல்வேறு பகுதிகளாக பிரிக்க, மாறியின் பெயர்களில் அடிக்கீருடன் கொள்ளலாம் (எ.கா: **total_sales** என்பது சரியான குறிப்பெயர்,**total sales** ஒரு தவறான குறிப்பெயர் ஆகும்.)
- **tamilMark** போன்ற பாணி கொண்ட மாறியை பயன்படுத்தலாம். இரண்டாவது வார்த்தையின் முதல் எழுத்து ஆங்கில 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 / மிதப்புப்புள்ளி மாறிலிகள் (Real constants / Floating point constants)

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

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.

Chapter 9.indd 121 4/11/2020 11:44:40 AM

(i) பதின்மம் (Decimal)

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

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 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 E1 → 5.864×10^1 → 58.64

5864 E-2 → 5864×10^{-2} → 58.64

0.5864 E2 → 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. சரநிலையுருக்கள் தானமைவாக 'Null' 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: "All", "Welcome" "1234"

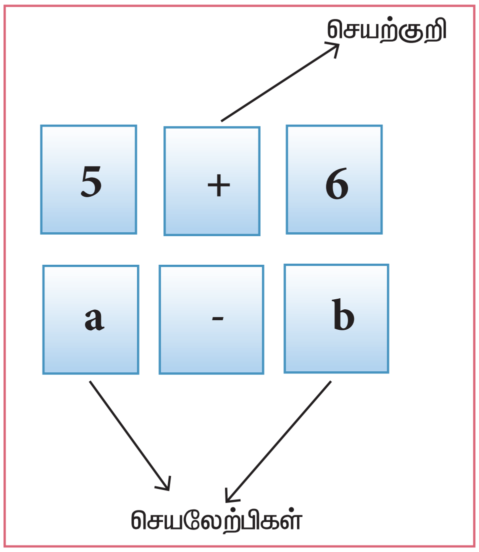
தவறான சரநிலையுருக்கள்: 'Welcome', '1234'

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. மதிப்பிருத்து செயற்குறிகள் (Assignment Operators)
5. நிபந்தனைச்செயற்குறிகள் (Conditional Operator)

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

Chapter 9.indd 124 4/11/2020 11:44:41 AM

(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.

- 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.

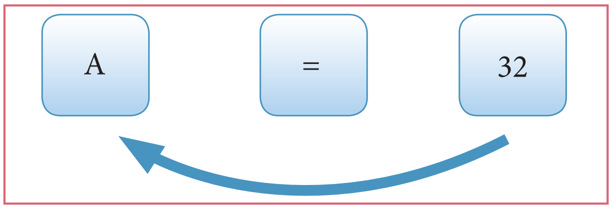
எடுத்துக்காட்டு: a = 5, b = 6, c = 7;

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) 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 125 4/11/2020 11:44:41 AM



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.

(5) 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.

(6) 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.
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,

Chapter 9.indd 126 4/11/2020 11:44:41 AM

செயற்குறிகளும் குறிப்பிட்ட தருக்க முறைப்படி குழுவாக்கப்பட்டு மதிப்பிடப்படுகின்றன. இத்தகைய குழுவாக்கம் **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

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 ( )

Chapter 9.indd 127 4/11/2020 11:44:41 AM

[ ] 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;
		}

சுய மதிப்பீடு

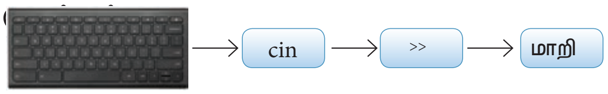
1. செயற்குறிகளின் பயன் யாது?
2. இரும செயற்குறி என்றால் என்ன? கணித இரும செயற்குறிகளுக்கு எடுத்துக்காட்டு தருக.
3. வகுமீதி (%) செயற்குறியின் செயல்பாடு என்ன?
4. 8.5 % 2 – விடை யாது?
5. R-ன் தொடக்க மதிப்பு 35 எனில் S=(R--)+(++R), என்ற கோவையில், S-யின் மதிப்பு என்ன?
6. தொடக்கத்தில் k=20 எனில் j=--k+2k என்ற 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



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.

Chapter 9.indd 128 4/11/2020 11:44:41 AM

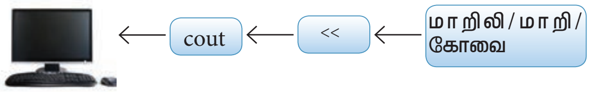
எடுத்துக்காட்டு:

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.

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.

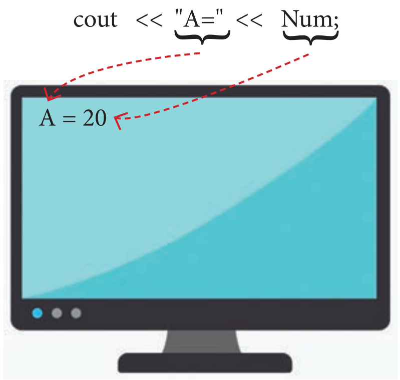
9.4.3. I/O செயற்குறிகள் தொடராக்கம் (Cascading of I/O operators)

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.



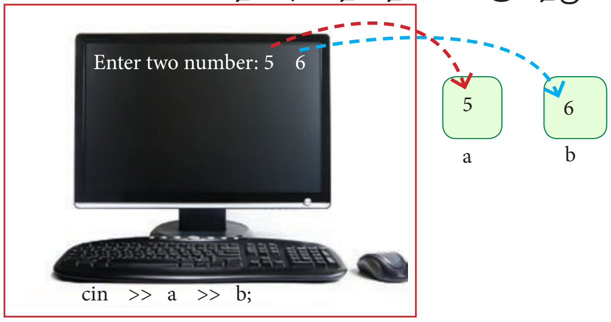
படம் 9.6 cout-ன் தொடராக்கம்

Chapter 9.indd 129 4/11/2020 11:44:41 AM

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.

Sample Prog 9_5.cpp

```
// C++ program to print a string
#include <iostream>
using namespace std;
int main()
{
    cout << "Welcome to Programming in C++";
    return 0;
}
```

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 தலைப்பு கோப்பை நிரலில் கொடுக்க தவறினால் cin மற்றும் cout –க்கான பிழைச் செய்தியை சுட்டிக்காட்டும் மற்றும் உள்ளீட்டைப் பெறவோ அல்லது வெளியீட்டை அனுப்பவோ முடியாது.
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)

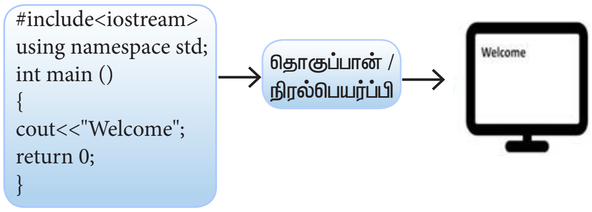
மூலக்குறிமுறையை தட்டச்சு செய்து முடித்த பின் கண்டிப்பாக .cpp என்ற நீட்டிப்பு பெயருடன் சேமிக்க வேண்டும்.

(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 பயனுருக்கு 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.



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

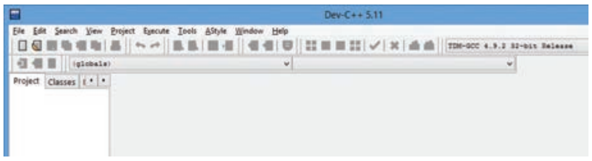
Chapter 9.indd 131 4/11/2020 11:44:42 AM

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 நுட்பங்களும் "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.

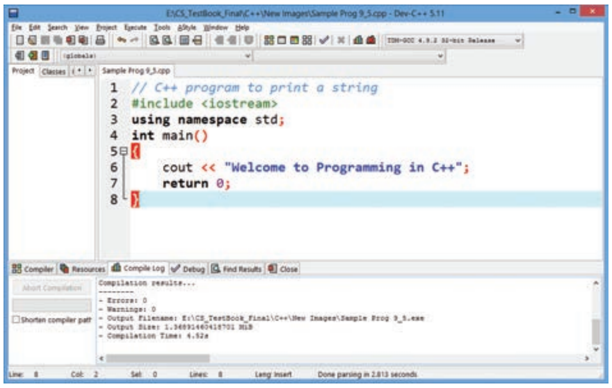
1. Dev C++ ஐ நிறுவிய பிறகு, Dev C++ பணிக்குறியானது திரைமுகப்பில் தோன்றும். IDE- யை திறப்பதற்கு இரட்டை கிளிக் செய்யவும். Dev C++ IDE பின்வருமாறு தோன்றும்.



படம் 9.9 Dev C++ தொடக்க சாளரம்

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

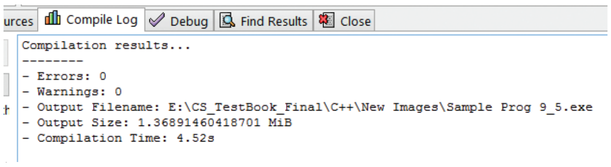
3. தோன்றுகின்ற திரையில் C++ நிரலை தட்டச்சு செய்து அதை சேமிக்க File → Save அல்லது **Ctrl + S** என்பதை பயன்படுத்த வேண்டும். இது தானமைவாக. cpp என்ற நீட்டிப்பை கோப்பின் பெயருடன் சேர்த்துக்கொள்ளும். கோப்புப் பெயருடன் .cpp என்பதை தட்டச்சு செய்யத் தேவையில்லை.



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

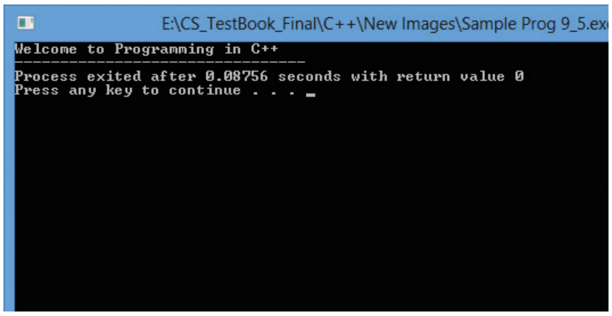
4. கோப்பினை சேமித்தப்பின், **Execute** → **Compile & Run** என்பதை கிளிக் செய்யவும் அல்லது F11 விசையை அழுத்தவும்.

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 132 4/11/2020 11:44:42 AM

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.

Chapter 9.indd 133 4/11/2020 11:44:42 AM

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;
}
```

- கொக்கப்படும் 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 134 4/11/2020 11:44:42 AM

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 மாஸ்சிட்டி
   (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 சரியனா சரநிலையுரு ஆகும்?
   (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
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

Chapter 9.indd 135 4/11/2020 11:44:42 AM

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.
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

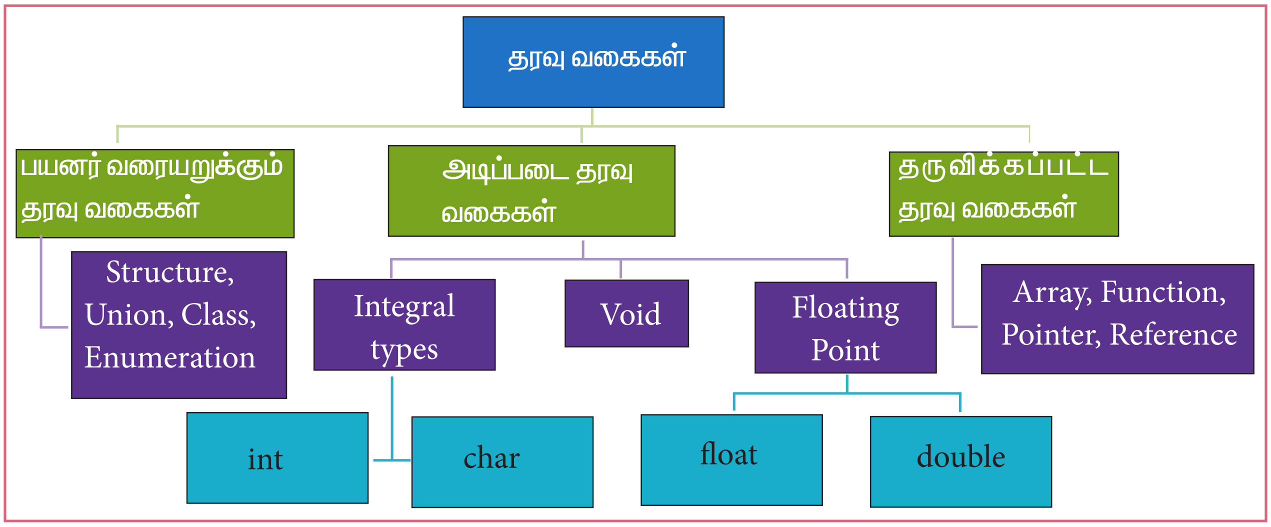
C++ல் தரவு இனங்கள் மூன்று வகைகளாக பிரிக்கப்படுகிறது. அவை,

(1) அடிப்படை தரவு இனங்கள் (Fundamental data types)

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

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

Chapter 9.indd 136 4/11/2020 11:44:42 AM



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 அறிவிக்க 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.

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 இயக்கப்படும்போதுபயனரால்கொடுக்கப்படுகின்ற மதிப்புகளை சேமிக்க num1, num2 என்ற மாறிகள் பயன்படுகின்றன. Sum என்ற மாறியானது செயல்படுத்தப்பட்ட மதிப்பை (விடை) சேமிக்கப் பயன்படுகிறது.

மேலே கொடுக்கப்பட்டுள்ள நிரல் இயங்கும் போது,வரிசைஎண் 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.

(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 எண்மதிப்பைசேமித்துக்கொள்ள 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, மிதப்பு புள்ளி 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.

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.

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

Chapter 9.indd 140 4/11/2020 11:44:43 AM

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	2 bytes	16 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 int 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++ நிரல் பெயர்ப்பியை 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++ நிரல் பெயர்ப்பியின் 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)

Chapter 9.indd 142 4/11/2020 11:44:43 AM

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
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 F என்ற எழுத்தை பயனபடுத்த வேண்டும்.

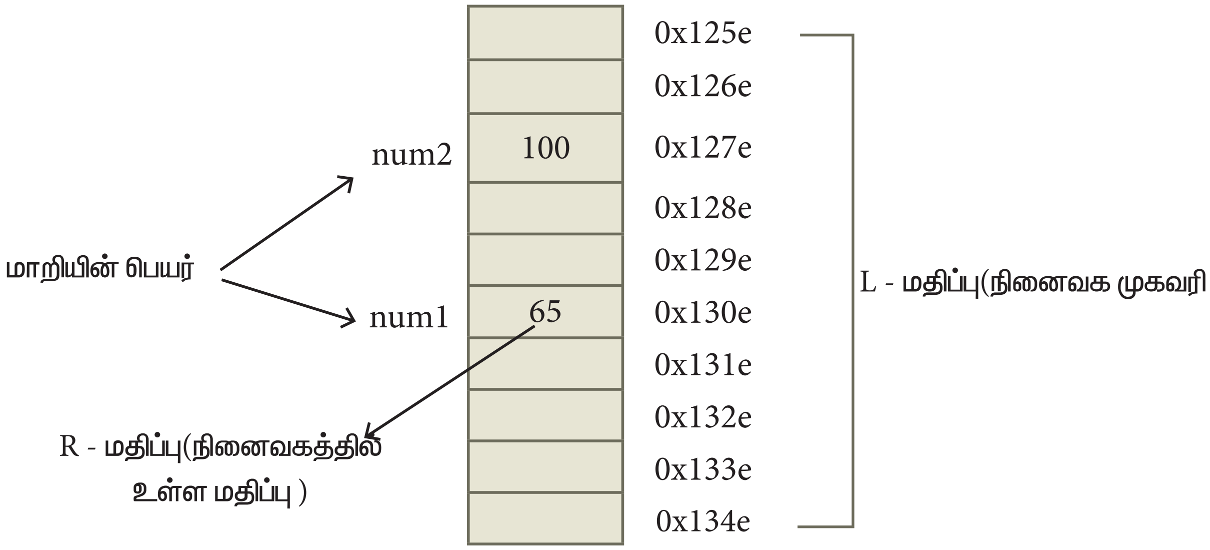
Chapter 9.indd 143 4/11/2020 11:44:43 AM

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

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 நிரல் பெயர்ப்பிக்கு தெரிவிப்பதே அறிவிப்பு ஆகும். எடுத்துக்காட்டாக, ஒரு மாறியை int வகையாக Dev C++ 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:

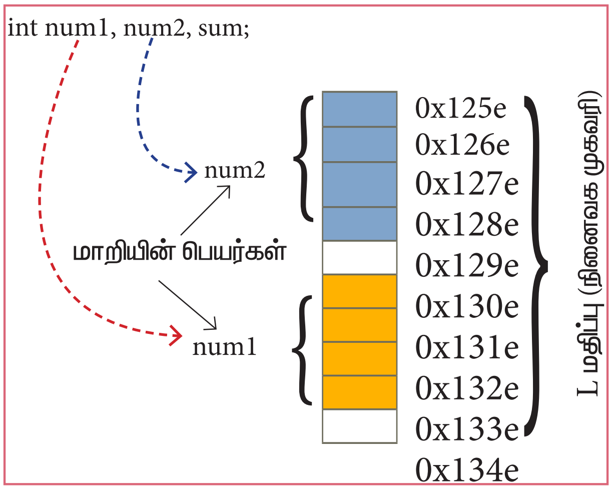
<data type> <var1>, <var2>, <var3> …… <var_n>;

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)

Chapter 9.indd 144 4/11/2020 11:44:43 AM



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;

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;

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.

Chapter 9.indd 146 4/11/2020 11:44:43 AM

```
#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.

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

வெளியீட்டை வடிவமைக்க C++ பல்வேறு கையாளுகைகளை வழங்கியுள்ளது. endl, setw, setfill, setprecision மற்றும் setf போன்றவை பொதுவாக பயன்படுத்தப்படும். கையாளுகை செயற்கூறுகளாகும். 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 <<**endl;**

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:

setw (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 and 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;
}
```

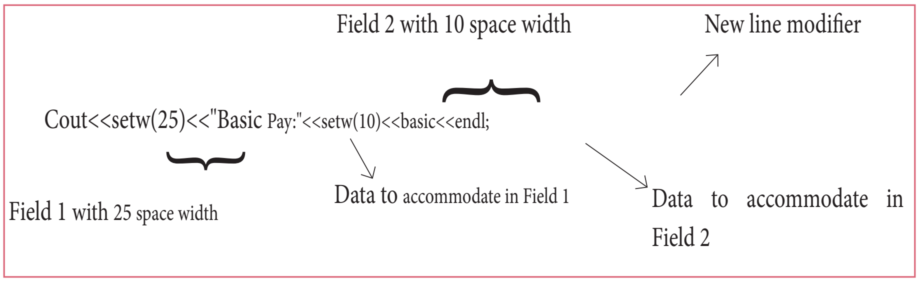

Chapter 9.indd 148 4/11/2020 11:44:43 AM

வெளியீடு:

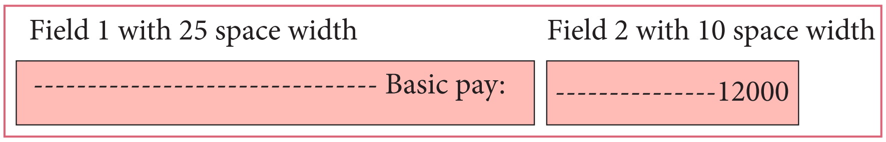
Enter Basic Pay: 12000
Enter D.A : 1250
Enter H.R.A: 1450

Basic Pay	:	12000
Dearness Allowance	:	1250
House Rent Allowance	:	1450
Gross Pay	:	14700
G.P.F	:	1325
Income Tax	:	1470
Net Pay	:	11905

மேலே கொடுக்கப்பட்ட நிரலில், ஒவ்வொரு வெளியீட்டுக் கூற்றும் இரண்டு setw( ) 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

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()
{
    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;
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. சுட்டு மாறிகள் * என்ற குறியுடன் அறிவிக்கப்படும்	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

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 வரையெல்லையை மீறினால் விடையை வரையறுக்க முடியாது.
3	long to short	தரவு இழப்பு

அட்டவணை 9.12 – வெளியுறை இனமாற்ற சிக்கல்கள்

```
#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. குறிப்பின் பயன் யாது?
5. Setprecision() –பயன் யாது?

Chapter 9.indd 152 4/11/2020 11:44:44 AM

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

மதிப்பீடு

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

Chapter 9.indd 153 4/11/2020 11:44:44 AM

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

பகுதி-இ

சிறு வினாக்கள் (3 மதிப்பெண்கள்) :

1. C++-ல் கணக்கீட்டுச் செயற்குறிகள் யாவை? ஒரும, இரும செயற்குறிகளை எடுத்துக்காட்டுடன் வேறுபடுத்துக.
2. ஒப்பீட்டு செயற்குறிகளும், தருக்கச் செயற்குறிகளும் எந்த வகையில் தொடர்புடையவை?
3. 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.