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 அறிமுகம் :

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] ;

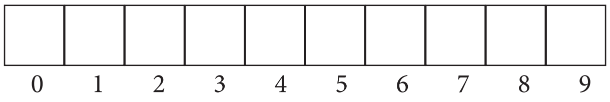
மேலே அறிவிக்கப்பட்டுள்ள “num” என்ற அணியானது 10 முழு எண் மதிப்புகளை சேமிக்கும் அணியாகும். (முழு எண் தரவு வகையைச் சார்ந்த

Chapter 12.indd 206 4/11/2020 11:48:10 AM

10 பல்வேறு மதிப்புகளை நினைவகத்தில் சேமிக்கலாம்).

கீழே உள்ள படத்தில் உள்ளதுபோல, இந்த அறிவிப்புக்கு "num" என்ற பொதுப் பெயரில் நிரல்பெயர்ப்பி 10 நினைவக இடங்களை (பெட்டிகள்) ஒதுக்கும்.

int num[10];



கீழொட்டுகள்

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.

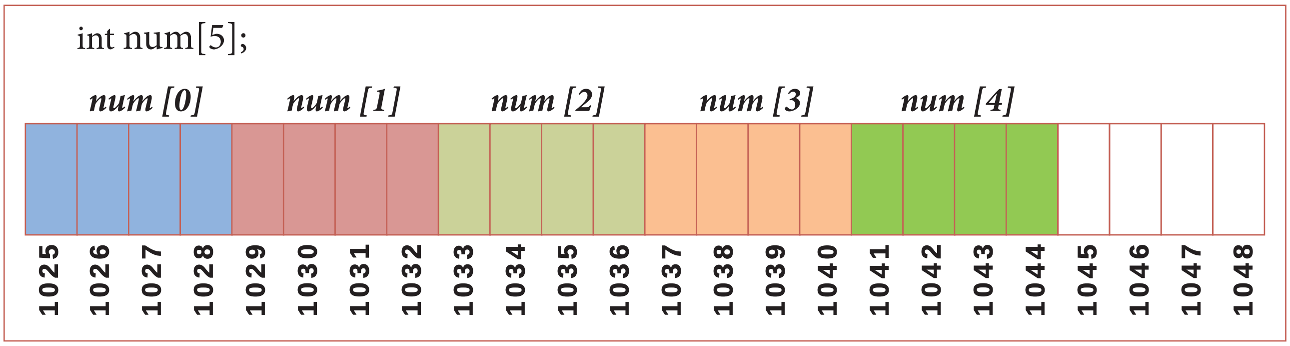
float salary[20];// salary என்ற அணி 20 தசமபுள்ளி உறுப்புகளை கொண்ட அணியாகும்.

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.



மேலே கொடுக்கப்பட்டுள்ள படத்தில், num என்ற அணியானது 5 உறுப்புகளைக் கொண்ட முழு எண் அணி என்பது காட்டப்பட்டுள்ளது. Dev - C++ நிரல் பெயர்ப்பியானது, 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
longdouble	10	10

அணியின் நினைவக ஒதுக்கீட்டை கணக்கிட பயன்படும் வாய்ப்பாடு

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};

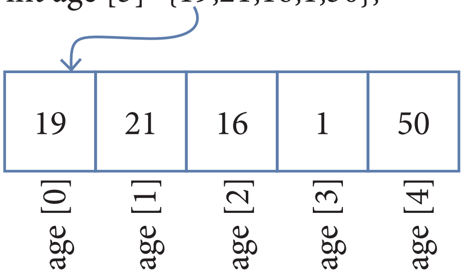
எடுத்துக்காட்டு

int age [5] = { 19,21,16,1,50};

Chapter 12.indd 207 4/11/2020 11:48:10 AM

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)

int age [5]={19,21,16,1,50};



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. அணியின் அளவு 5 என எடுத்துக்கொள்ளப்படும்

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'};

நிரலை இயக்கும்போது அணிக்குறிய மதிப்புகளை உள்ளீடு செய்தல்:

நிரலின் இயக்க நேரத்தில் அணிக்கு ஒரு மதிப்பை மட்டுமே உள்ளீடு செய்ய முடியும். ஒன்றிற்க்கு மேற்பட்ட மதிப்புகளை உள்ளீடு செய்ய வேண்டுமெனில் ஏதேனும் ஒரு மடக்கைப் பயன்படுத்தலாம். for மடக்கானது அணியின் உறுப்புகளுக்கு மதிப்பைப் 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.

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.

Chapter 12.indd 208 4/11/2020 11:48:10 AM

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.

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.)

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
```

(HOT மேலே கொடுக்கப்பட்டுள்ள நிரலை if கூற்றிற்கு பதிலாக நிபந்தனைச் செயற்குறியைப் பயன்படுத்தி எழுதவும்)

ஒரு பரிமாண அணியில் தேடல்

Chapter 12.indd 210 4/11/2020 11:48:10 AM

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 main()
{
        int num[10], val, id=-1;
        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;
     for (int i=0; i<size; i++)
     {
          if (arr[i] == value)
           {    id= i;
                break;
           }
     }
        if(id==-1)
        cout<< "\n Given value is not found in the array..";
        else
        cout<< "\n The value is found at the position" << id+1;
        return 0;
}
```

மேலே கொடுக்கப்பட்டுள்ள நிரல் தேடப்படும் மதிப்பை பட்டியலில் உள்ள உறுப்புகளின் மதிப்புடன் ஒப்பிட்டு அந்த மதிப்பை கண்டுபிடிக்கும்வரை அணியிலுள்ள அனைத்து உறுப்புகளுக்கும் பயணித்து ஒப்பிடும்.

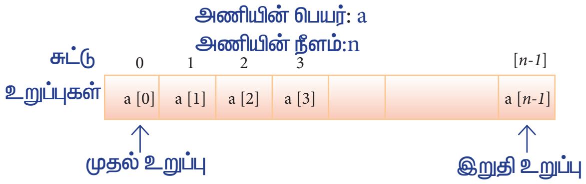
சரங்கள்

குறியுருக்களின் வரிசையை 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).

Chapter 12.indd 211 4/11/2020 11:48:10 AM



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.

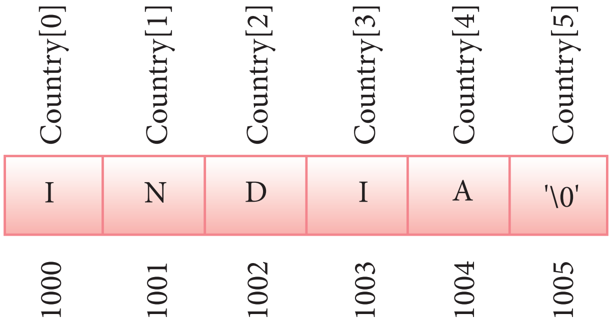
char arrayName[size]={ list of characters separated by comma } ;

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.

Chapter 12.indd 212 4/11/2020 11:48:11 AM



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 வழிகள் 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

குறிப்பு

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 is too long" என்ற பிழை அறிக்கையை வெளியிடும்.

கொடுக்கப்படும் சொல்லானது palindrome சொல்லா என சோதிக்க C++ நிரல் எழுதுதல்.

```
#include<iostream>
using namespace std;
int main( )
{
        int i, j, len, flag =1;
        char a [20];
        cout<<"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)
                cout<<"\n The String is palindrome";
        else
                cout<<"\n The String is not palindrome";
        return 0;
}
```

வெளியீடு:

```
        Enter a string : madam
        The String is palindrome
```

12.3 இரு பரிமாண அணி

இரு பரிமாண அணி என்பது ஒரே தரவினத்தை சார்ந்த உறுப்புகளின் தொகுதியாகும். 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

Chapter 12.indd 214 4/11/2020 11:48:11 AM

12.3.1 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 தகுதிவாய்ந்த 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

சரங்களின் உறுப்புகளுக்கு தொடக்க மதிப்பிருத்தும்போது, அதன் அளவிற்கு அதிகமாக மதிப்புகளை இருத்தக்கூடாது.

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.

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 [1] [2] =60; // இரண்டாம் வரிசையில் உள்ள மூன்றாம் உறுப்பிற்கு 60 என்ற மதிப்பிருத்தும்

matrix [3] [0] =100; // முதல் வரிசையில் உள்ள நான்காம் உறுப்பிற்கு 100 என்ற மதிப்பிருத்தும்.

இரு அணிக்கோவைகளில் கூட்டுத்தொகையை கணக்கிட தேவையான நிரல்

Chapter 12.indd 215 4/11/2020 11:48:11 AM

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>>m2[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;
}
```


Chapter 12.indd 216 4/11/2020 11:48:11 AM

```
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];

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			

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

Chapter 12.indd 217 4/11/2020 11:48:11 AM

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.

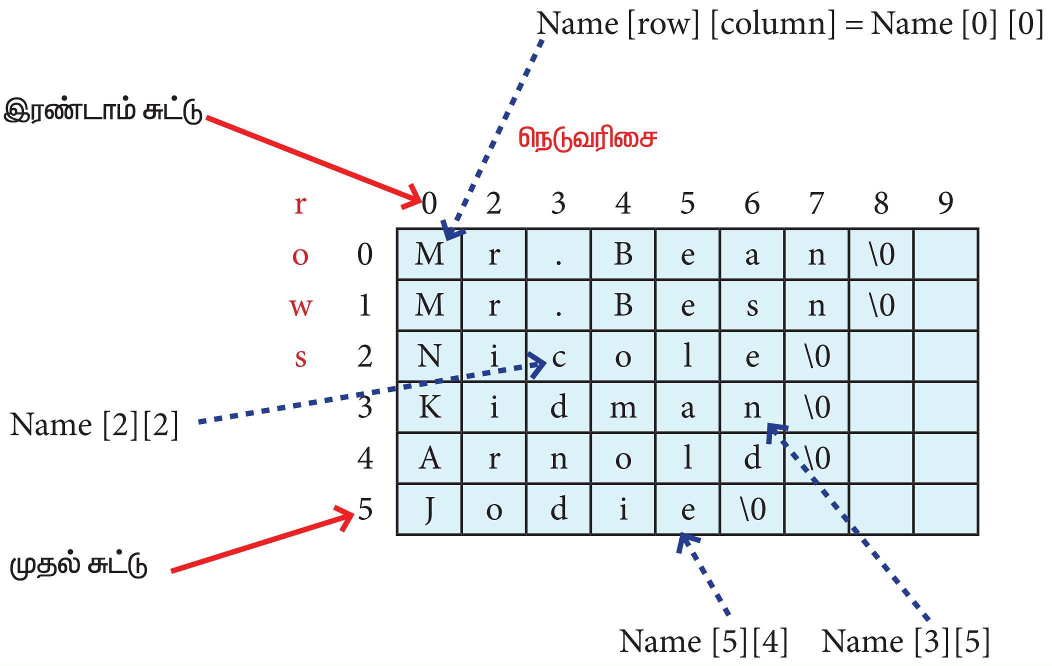
12.4.1 தொடக்கமதிப்பிருத்துதல்

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 கட்டுருக்கள் அறிமுகம்

வெவ்வேறு வகையான தரவு இனங்களை கொண்ட பயனர் வரையறுக்கும் தரவினம் கட்டுரு எனப்படும். இது பல்வேறு வகையான தரவு இனங்களுடன் கூடிய மாறிகளை ஒரே தொகுதிக்குள் ஒன்றிணைத்துள்ளது.

12.5.1. கட்டுருக்களின் நோக்கம்

ஒரே மாதிரியான தரவு வகைகளை சார்ந்த ஒன்றுக்கு மேற்பட்ட மாறிகள் தேவைப்படும் போது அவற்றைக் 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 தொகுதியில் அறிவித்து, அவைகளுக்கு நினைவகத்தில் அருகருகே இடம் ஒதுக்க கட்டுரு உதவுகிறது.

12.5.2 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
{
    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.

Chapter 12.indd 219 4/11/2020 11:48:11 AM

```
struct Student
{
long rollno;
int age;
float weight;
}balu, frank;
```

12.5.3 கட்டுரு உறுப்புகளை அணுகுதல்

student என்ற கட்டுரு தரவினத்தில் இரண்டு பொருள்களை அறிவித்தவுடன் அவறிற்கான தரவு உறுப்புகளை நேரடியாக அணுகலாம். பொருளானது தரவு உறுப்புகளை அணுகுவதற்கு, பொருளின் பெயர் மற்றும் இரண்டு பொருள்களை அறிவித்தவுடன் 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
```

தரவு உறுப்பானது கட்டுறு வகையாக இருந்தால் → உறுப்புகளை அணுகப்பயன்படுகிறது. name என்பது student கட்டுருவின் குறியுரு சுட்டு (Char *name) என அறிவிக்கப்பட்டால் student→ name என்பதன் மூலம் அதை அணுக முடியும்.

> 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.

12.5.4 கட்டுரு உறுப்புகளுக்கு தொடக்க மதிப்பிருத்தல் (Initializing Structure elements)

மாறிகளுக்கு மதிப்புகள் இருத்தப்படுவதை போன்றே கட்டுரு உறுப்புகளுக்கும் மதிப்பு இருத்தப்படும்.

(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};

12.5.5 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.

```
struct Student
{
        int  age;
        float height, weight;
}mahesh;
```

> இரண்டு கட்டுரு மாறிகள் (அ) பொருள்கள் ஒரே வகையாக இருந்தால் மட்டுமே கட்டுரு மதிப்பிருத்துதலை செய்ய இயலும்.

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
```

நினைவில் கொள்க

- கட்டுரு என்பது பல்வேறு வகையான தரவின உறுப்புகளைக் கொண்ட பயனரால் வரையறுக்கப்பட்ட தரவினம் ஆகும்.
- கட்டுருவை வரையறுக்க struct 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.

ஆய்வு அறிக்கை

1. 10 மாணவர்களின் மதிப்பெண்களை உள்ளீடாக பெற்று அவற்றின் சராசரி அவற்றுள் அதிகமான மதிப்பெண் மற்றும் குறைவான மதிப்பெண்களை காண ஒரு நிரலை எழுதுக.

மதிப்பீடு

N1R5E4

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. int age[]={6,90,20,18,2}; இந்த அணியில் எத்தனை உறுப்புகள் உள்ளன?
   அ) 2 ஆ) 5 இ) 6 ஈ) 4
3. cin>>n[3]; இந்த கூற்று எந்த உறுப்பில் மதிப்பை உள்ளீடும்?
   அ) 2 ஆ) 3 இ) 4 ஈ) 5
4. சரங்கள் தானமைவாக இவற்றுள் எந்த குறியுருவுடன் முடிவடையும்?
   அ) \0 ஆ) \t இ) \n ஈ) \b
5. கட்டுரு வரையறை எந்த செயற்குறியுடன் முடிவடைதல் வேண்டும்?
   (அ) : (ஆ) } (இ) ; (ஈ) ::

Chapter 12.indd 222 4/11/2020 11:48:11 AM

6. கட்டுருக்களை அறிவிக்கும் போது என்ன ஏற்படும்?

(அ) அது எந்த நினைவகத்தையும் ஒதுக்காது

(ஆ) அது நினைவகத்தை ஒதுக்கும்

(இ) அது அறிவிக்கும் மற்றும் தொடங்கும்

(ஈ) அது அறிவிக்க மட்டும் செய்யும்

7. 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

8. கீழ்கண்டவற்றுள் எவை சரியான கட்டுரு வரையறை?

(அ) struct {int num;} (ஆ) struct sum {int num;}

(இ) struct sum int sum; (ஈ)struct sum {int num;};

9. 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;

10. கட்டுரு உறுப்புகளை அணுகும் போது புள்ளி செயற்குறியின் வலது புறமுள்ள குறிப்பெயரின் பெயர்

(அ) கட்டுறு மாறி structure variable (ஆ) கட்டுறு பெயர்

(இ) கட்டுறு உறுப்பு (ஈ) கட்டுறு செயற்கூறு

பகுதி– ஆ

II குறுவினாக்கள் (2 மதிப்பெண்கள்)

1. அணியில் பயணித்தல் என்றால் என்ன?
2. சரங்கள் என்றால் என்ன?
3. இரு பரிமாண அணியை அறிவிக்கும் தொடரியலை எழுதுக.
4. வரையறு-கட்டுரு. அதன் பயன் என்ன?
5. பின்வரும் கட்டுரு வரையறையில் பிழை என்ன?

```
struct   employee{ inteno;charename[20];char dept;}

Employee e1,e2;
```

பகுதி– இ

III சிறுவினாக்கள் (3 மதிப்பெண்கள்)

1. அணி என்றால் என்ன ? அதன் வகைகளை எழுதுக.
2. சரங்களின் அணியைப் பற்றி சிறுகுறிப்பு எழுதுக.
3. 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;
}
```

4. ஒரு கட்டுருவின் உறுப்புகளை எவ்வாறு அணுக முடியும்? எடுத்துக்காட்டு தருக.
5. பெயரற்ற கட்டுரு என்றால் என்ன? எடுத்துக்காட்டு தருக.

பகுதி – ஈ

IV பெருவினாக்கள் (5 மதிப்பெண்கள்)

1. இரண்டு அணிக்கோவைகளில் உள்ள மதிப்புகளின் வித்தியாசம் கண்டறிய நிரலை எழுதுக.
2. பின்வரும் கட்டுரு வரையறையை பயன்படுத்தி இரண்டு தூரங்களை (distance) கூட்டுவதற்கான c++ நிரலை எழுதுக.

```
struct Distance{

int feet;

float inch;

}d1 , d2, sum;
```

3. 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;
}
```

4. 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;
```

```
        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;
}
```

5. 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)
{
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