பாடம் 15

N28ASK

பல்லுருவாக்கம்

கற்றலின் நோக்கங்கள் :

இந்த பாடப்பகுதியை கற்றபின் மாணவர் அறிந்து கொள்வது

- பணிமிகுப்பு பற்றி புரிந்து கொள்ளுதல்.
- செயற்கூறு பணிமிகுப்பு, ஆக்கி பணிமிகுப்பு, செயற்குறி பணிமிகுப்பு கொண்ட C++ நிரல்களை உருவாக்குதல்.
- பல்லுருவாக்க கருத்துகளைக் கொண்டு நிரல்களை இயக்குதல் மற்றும் பிழைதிருத்தம் செய்தல்.

15.1 அறிமுகம்

பல்லுருவாக்கம் (polymorphism) என்ற சொல் பல வடிவங்கள் (poly – many, morph – shapes) என்னும் பொருளைத் தருகிறது. பல்லுருவாக்கம் என்பது ஒரு பொருள் அல்லது செயற்கூறினை பல்வேறு வடிவங்களில் காண்பிக்க உதவுகிறது.

15.2 செயற்கூறு பணிமிகுப்பு (Function overloading)

செய்தி அல்லது தரவினை ஒன்றுக்கு மேற்பட்ட வடிவங்களில் செயலாக்கவல்ல செயற்கூறின் திறனையே செயற்கூறு பணிமிகுப்பு என்கிறோம். வேறு வகையில் கூறினால், பணி மிகுப்பு என்பது இரண்டு அல்லது அதற்கு மேற்பட்ட செயற்கூறுகள் ஒரே பெயரைப் பகிர்ந்து கொண்டு வேறுபட்ட அளபுருக்களை கொண்டிருக்கும். செயற்கூறுகள் ஒரே பெயரை பகிர்ந்து கொள்ளுதலை பணிமிகுப்பு என்றும் இந்த செயல்பாட்டை செயற்கூறு பணிமிகுப்பு என்கிறோம். செயற்கூறின் அளபுருக்களின் எண்ணிக்கை மற்றும் தரவினங்களை செயற்கூறு முன்வடிவு என்கிறோம். பணி மிகுக்கப்பட்ட செயற்கூறினை அழைக்கும் போது, நிரல் பெயர்ப்பி மிகச் சரியான வரையறுப்பை, அழைக்கப்பட்ட செயற்கூறின் அளபுருக்களின் வகையோடு வரையறுக்கப்பட்ட செயற்கூறின் செயலுருபுகளின் வகையோடு ஒப்பிட்டு தீர்மானிக்கும். மிகச் சிறந்த செயற்கூறு அல்லது செயற்குறி பணிமிகுப்பு தேர்ந்தெடுப்பு முறையை பணிமிகுப்பு தீர்மானம் என்கிறோம்.

15.2.1 செயற்கூறு பணிமிகுப்பின் தேவைகள் (Need For Function overloading)

சில சமயங்களில், ஒரே செயலைக் குறிக்கும் வெவ்வேறு பொருள் கொண்ட பெயரை க்கண்டறிதல் கடினமான செயல் ஆகும்.

வட்டம், முக்கோணம் மற்றும் செவ்வகம் ஆகியவற்றின் பரப்பளவைக் கண்டறியும் செயற்கூறு வரையறைகளைக் கீழே காணலாம்.

float area_circle(float radius) // ஒரு வட்டத்தின் பரப்பைக் கணிக்க

float area_triangle(float half,floatbase,float height) // முக்கோணத்தின் பரப்பைக் கணிக்க

float area_rectangle(float length , float breadth) // செவ்வகத்தின் பரப்பைக் கணிக்க

இதனைக் கீழ்கண்ட விதத்தில் ஒற்றை செயற்கூறு தலைப்பைக் கொண்டு மாற்றி எழுதலாம்.

float area (float radius);

float area (float half, float base, float height);

float area (float length , float breadth);

Chapter 15.indd 248 4/11/2020 11:49:44 AM

நிரல்15.1 செயற்கூறு பணிமிகுப்பினை விளக்கும் எளிய C++ நிரலை கீழே காணலாம்.

```
#include <iostream>
using namespace std;
void print(int i)
 {cout<< " It is integer " << i <<endl;}
void print(double  f)
{  cout<< " It is float " << f <<endl;}
void print(string c)
{  cout<< " It is string " << c <<endl;}
int main() {
            print(10);
            print(10.10);
            print("Ten");
            return 0;
}
```

வெளியீடு

```
            It is integer 10
            It is float 10.1
            It is string Ten
```

குறிப்பு:-

செயற்கூறு பணிமிகுப்பு, பல்லுருவாக்கத்தை மட்டுமே நடைமுறைப்படுத்தாமல் ஓர் நிரலில் ஒப்பீடுகளின் எண்ணிக்கையைக் குறைத்து, நிரல் வேகமாக செயல்பட உதவுகிறது. நிரலர், அதிக செயற்கூற்றின் பெயர்களை நினைவில் வைத்துக் கொள்வதை தவிர்க்க வழி செய்கிறது.

15.2.2 செயற்கூறு பணிமிகுப்பிற்கான விதிமுறைகள் (Rules for function overloading)

1. பணிமிகுத்த செயற்கூறுகள் முறையான அளபுருக்களின் எண்ணிக்கையிலோ, அல்லது அவற்றின் தரவு இனங்களிலோ வேறுபட்டிருக்க வேண்டும்.
2. பணிமிகுத்த செயற்கூறுகள் திருப்பியனுப்பும் தரவினம் ஒன்றாக இருக்க வேண்டும் என்ற தேவையில்லை.
3. பணிமிகுத்த செயற்கூறுகளின் தானமைவு செயலுருபுகளை அளபுருக்களின் பட்டியலில் ஒரு பகுதியாக C++ நிரல் பெயர்ப்பி கருதிக் கொள்ளாது.

நிரல் 15.2 செயற்கூறு பணிமிகுப்பினை விளக்கும் C++ நிரலை கீழே காணலாம்

```
#include <iostream>
using namespace std;
long add(long, long);
long add(long,long,long);
float add(float, float);
int main()
{
 long a, b, c,d;
 float e, f, g;
 cout << "Enter three integers\n";
cin >> a >> b>>c;
d=add(a,b,c);   //number of arguments different  but  same data type
cout << "Sum of 3 integers: " << d << endl;
cout << "Sum of 3 integers: " << d << endl;
cout << "Enter two integers\n";
cin >> a >> b;
c = add(a, b);   //two arguments data type same with above function call and
different with below function call
cout << "Sum of 2 integers: " << c << endl;
cout << "Enter two floating point numbers\n";
```


Chapter 15.indd 249 4/11/2020 11:49:44 AM

```
cin >> e >> f;
 g = add(e, f);  //two arguments  similar to the above function call but data
type different
cout << "Sum of floats: " << g << endl;
}
long add(long c, long g)
{
 long sum;
 sum = c + g;
 return sum;
}
float add(float c, float g)
{
 float sum;
 sum = c + g;
 return sum;
}
long add(long c, long g,long h)
{
 long sum;
 sum = c + g+h;
 return sum;
}
```

வெளியீடு

```
Enter three integers
3 4 5
Sum of 3 integers: 12
Enter two integers
4 6
Sum of 2 integers: 10
Enter two floating point numbers
2.1 3.1
Sum of floats: 5.2
```

15.3 ஆக்கி பணிமிகுப்புConstructor overloading

இனக்குழுவின் சிறப்பு செயற்கூறுகளான ஆக்கிகளையும், செயற்கூறு பணிமிகுப்பு செய்ய முடியும். ஓர் இனக்குழுவில் வெவ்வேறு வரையறுப்புகளைக் கொண்ட ஒன்றுக்கு மேற்பட்ட ஆக்கிகள் இடம் பெறலாம். ஓர் இனக்குழுவிற்கு, பல்வேறு வகையான பொருள்களை உருவாக்க ஆக்கி பணிமிகுப்பு வழி வகை செய்கிறது.

Chapter 15.indd 250 4/11/2020 11:49:44 AM

நிரல் 15.3 ஆக்கி பணிமிகுப்பை விளக்கும் நிரல்

```
#include<iostream>
using namespace std;
class add
{ int num1, num2, sum;
public:
add()
{       cout<<"\n Constructor without parameters.. ";
num1= 0;  num2= 0;  sum = 0; }
add ( int s1, int s2 )
{ cout<<"\n Parameterized constructor... ";
 num1= s1;  num2=s2; sum=0; }
add (add &a)
{ cout<<"\n Copy Constructor ... ";
 num1= a.num1;
 num2=a.num2;
 sum = 0; }
void getdata()
{ cout<<"\nEnter data ... "; cin>>num1>>num2; }
void addition()
{ sum=num1+num2; }
void putdata() {
cout<<"\n The numbers are..";
cout<<num1<<'\t'<<num2;
cout<<"\n The sum of the numbers are.. "<< sum; } };
int main() {
add a, b (10, 20) , c(b);
a.getdata();
a.addition();
b.addition();
c.addition();
cout<<"\n Object a : ";
a.putdata();
cout<<"\n Object b : ";
b.putdata();
cout<<"\n Object c.. ";
c.putdata();
return 0; }
```

Compiler identifies a given member function is a constructor by its name and the return type.

Output

```
Constructor without parameters..
 Parameterized constructor...
 Copy Constructor ...
Enter data ...  20 30
 Object a :
 The numbers are..20    30
 The sum of the numbers are.. 50
 Object b :
 The numbers are..10    20
 The sum of the numbers are.. 30
 Object c..
 The numbers are..10    20
 The sum of the numbers are.. 30
```

குறிப்பு

ஆக்கிகள் பல இடம் பெற்றிருப்பதால், ஒரு பொருள் உருவாக்கப்படும் போதே செயலுருபுகள் அவற்றிற்கு அனுப்பி வைக்கப்பட வேண்டும்.

15.4 செயற்குறி பணிமிகுப்பு

செயற்குறி பணிமிகுப்பு என்பது +,++,-,—,+=,-=,*.<,> போன்ற வழக்கிலுள்ள C++ செயற்குறிகளுக்கு கூடுதலான செயல்பாடுகளை வரையறுப்பதைக் குறிக்கிறது. இதுவும் ஒரு பல்லுருவாக்க செயல் எனலாம். ஏனெனில், இதில் செயற்குறி பணிமிகுக்கப்பட்டு, செயற்குறிக்கு நிரலர் விரும்புகிற பொருளை வழங்குகிறது.

எடுத்துக்காட்டாக '+' என்ற செயற்குறியை பணிமிகுப்பு முழு எண்ணிற்கு கூடுதலையும், சரத்திற்கு இணைத்தல் போன்ற செயல்பாட்டினையும் பல்வேறு தரவினங்களுக்கு பயன்படுத்தி பணிமிகுப்பு செய்யலாம். C++ மொழியில், சில செயற்குறிகளை தவிர்த்து, அனைத்து செயற்குறிகளையும் பணி மிகுப்பு செய்ய முடியும்.

பணிமிகுப்பு செய்ய முடியாத செயற்குறிகள் பின்வருமாறு

- வரையெல்லை செயற்குறி (::)
- sizeof செயற்குறி
- உறுப்பு தேர்வி (member selector - .)
- உறுப்பு சுட்டல் தேர்வி (member pointer selector - *)
- நிபந்தனை செயற்குறி (conditional operator - ?:)

செயற்குறி பணிமிகுப்பின் கட்டளையமைப்பு (Operator Overloading Syntax)

சிறப்புச் சொல்

பணிமிகுக்கப்பட வேண்டிய செயற்குறி

```
ReturnType classname :: Operator Operator Symbol (செயலுருபுகளின் பட்டியல்)
{
        \\ Function body
}
```

15.4.1 செயற்குறி பணிமிகுப்பின் வரம்பெல்லைகள் (Restrictions on Operator Overloading)

செயற்குறி பணிமிகுப்பினை நடைமுறைப்படுத்தும் போது பின்வரும் வரம்பெல்லைகளைக் கவனத்தில் கொள்ள வேண்டும்.

1. ஒரு செயற்குறியின் முன்னுரிமையும், திசைமுகத்தையும் மாற்ற இயலாது.

2. புதிய செயற்குறிகளை உருவாக்க முடியாது. ஏற்கனவே இருக்கும் செயற்குறிகளை மட்டுமே பணிமிகுக்க முடியும்.

3. ஒரு செயற்குறியின் அடிப்படை செயல் முறையை மறுவரையறை செய்ய முடியாது. முழு எண்கள் கூட்டப்படும் முறையை மாற்றி அமைக்க முடியாது, ஆனால் கூடுதல் செயல்பாட்டினை அந்த செயற்குறிக்கு வழங்கலாம்.

4. பணிமிகுக்கப்பட்ட செயற்குறிகள் தானமைவு செயலுருபுகளைக் கொண்டிருக்காது.

5. இரும செயற்குறிகளை பணிமிகுக்கும் போது, அச்செயற்குறியின் இடப்பக்கம் அமையும் பொருள், அது வரையறுக்கப்பட்டுள்ள இனக்குழுவின் பொருளாக இருக்க வேண்டும்.

நிரல் 15.6 '+' குறியீடு பயன்படுத்தி இரும செயற்குறி பணிமிகுத்தலை விளக்கும் எடுத்துக்காட்டு நிரல்

```
#include<iostream>
using namespace std;
class complex
{ int real,img;
   public:
void read()
{
cout<<"\nEnter the REAL PART : ";
cin>>real;
cout<<"\nEnter the IMAGINARY PART : ";
cin>>img;
}
complex operator +(complex c2)
{
    complex c3;
    c3.real=real+c2.real;
    c3.img=img+c2.img;
    return c3;
}
    void display()
{ cout<<real<<"+"<<img<<"i"; }
};
int main()
{
    complex c1,c2,c3;
    int choice, cont;
    cout<<"\n\nEnter the First Complex Number";
    c1.read();
    cout<<"\n\nEnter the Second Complex Number";
    c2.read();
    c3=c1+c2; // binary + overloaded
    cout<<"\n\nSUM = ";
    c3.display();
    return 0;
 }
```

வெளியீடு

```
Enter the First Complex Number
Enter the REAL PART : 3
Enter the IMAGINARY PART : 4
Enter the Second Complex Number
Enter the REAL PART : 5
Enter the IMAGINARY PART : 8
SUM = 8+12i
```

நிரல் 15.7 செயற்குறி பணிமிகுப்பினைப் பயன்படுத்தி சரங்களை இணைக்கும் எடுத்துக்காட்டு நிரல்

```
#include<string.h>
#include<iostream>
using namespace std;
class strings
{
        public:
        char s[20];
        void getstring(char str[])
{
        strcpy(s,str);
  }
        void operator+(strings);
};
void strings::operator+(strings ob)
{
        strcat(s,ob.s);
        cout<<"\nConcatnated String is:"<<s;
}
int main()
{
        strings ob1, ob2;
        char string1[10], string2[10];
        cout<<"\nEnter First String:";
        cin>>string1;
        ob1.getstring(string1);
        cout<<"\nEnter Second String:";
        cin>>string2;
        ob2.getstring(string2);
        //Calling + operator to Join/Concatnate strings
        ob1+ob2;
        return 0;
}
```

வெளியீடு

```
Enter First String:COMPUTER
Enter Second String:SCIENCE
Concatnated String is:COMPUTERSCIENCE
```


Chapter 15.indd 254 4/11/2020 11:49:44 AM

நினைவில் கொள்க:

- C++ மொழியில் பல்லுருவாக்கம் செயற்கூறு பணிமிகுப்பு மற்றும் செயற்குறி பணிமிகுப்பு ஆகியவற்றின் மூலம் நிறைவேற்றப்படுகிறது.
- பணிமிகுப்பு என்பது ஒரே பெயர், ஒன்றுக்கு மேற்பட்ட பல்வேறு பொருளைக் குறிக்கிறது.
- பணிமிகுத்த செயற்கூறு என்பது ஒன்றுக்கு மேற்பட்ட தனித்த பொருள் கொண்ட செயற்கூறினை குறிக்கிறது.
- பணிமிகுத்த செயற்கூறுகள் ஒரே பெயரைக் கொண்டிருக்கும். ஆனால் முன்வடிவு வேறுபட்டிருக்கும் (அளபுருக்களின் எண்ணிக்கை மற்றும் தரவினம்)
- செயற்கூறின் அளபுருக்களின் பட்டியல் செயற்கூறு முன்வடிவு எனப்படுகிறது.
- இரண்டு செயற்கூறுகளில் ஒன்று குறிப்பு அளபுருக்களையும் மற்றொன்று மதிப்பு அளபுருக்களையும் கொண்டிருந்தால் அவற்றை பணிமிகுக்க முடியாது.
- சாதாரண செயற்கூறுகளைப் போல் உறுப்பு செயற்கூறுகளையும் பணிமிகுக்க முடியும்.
- ஓர் இனக்குழு பணிமிகுக்கப்பட்ட ஆக்கிகளை கொண்டிருக்க முடியும். அழிப்பிகள் பணிமிகுக்கப்பட முடியாது.
- ஒரு செயற்குறிக்குப் புதிய பொருளை வழங்கும் செயல்நுட்பமே செயற்குறி பணிமிகுப்பு என்றழைக்கப்படுகிறது.
- செயற்குறி பணிமிகுப்பு C++ செயற்குறிகள் பல வற்றிற்கு புதிய வரையறுப்புகளை அளிக்கிறது.
- பயனர் வரையறுக்கும் இனங்களையும் (பொருள்களைப்) பணிமிகுக்க முடியும்.
- பணிமிகுக்கப்பட்ட செயற்குறியின் வரையறை இவ்வாறு அமையும். சிறப்பு சொல் “Operator” அடுத்து செயற்குறி குறியீடு அமைந்திருக்கும்.
- கீழ்கண்ட C++ செயற்குறிகளை தவிர்த்து, அனைத்து செயற்குறிகளையும் பணிமிகுக்க முடியும்.
- இனக்குழு உறுப்பு அணுகல் செயற்குறி (.)
- வரையெல்லை செயற்குறி (::),
- sizeof () செயற்குறி
- நிபந்தனை செயற்குறி (?:)

மதிப்பீடு

பகுதி -அ

சரியான விடைகளைத் தேர்ந்தெடுத்து எழுதுக:

1. பின்வருவனவற்றுள் எது செயற்கூறுகளுக்கு வேறுபட்ட பொருள் உள்ளதை குறிக்கிறது?

   (அ) செயற்கூறு பணிமிகுப்பு

   (ஆ) உறுப்பு பணிமிகுப்பு

   (இ) செயற்குறி பணிமிகுப்பு

   (ஈ) செயற்பாடு பணிமிகுப்பு

2. பின்வருவனவற்றுள், எது நிரலின் ஒப்பீடுகளின் எண்ணிக்கையை குறைக்கிறது ?

   (அ) செயற்கூறு பணிமிகுப்பு

   (ஆ) செயற்பாடு பணிமிகுப்பு

   (இ) செயற்குறு பணிமிகுப்பு

   (ஈ) உறுப்பு பணிமிகுப்பு

3. $ என்ற குறியீட்டை 10 முறை வெளியிட கீழ்காணும் நிரலில் dispchar() என்ற செயற்கூறை எவ்வாறு அழைப்பாய்?

```
void dispchar(char ch='$',int size=10)
{
for(int i=1;i<=size;i++)
cout<<ch;
}
```

   (அ) dispchar();

   (ஆ) dispchar(ch,size);

   (இ) dispchar($,10);

   (ஈ)dispchar('$',10 times);

N29HHE

Chapter 15.indd 255 4/11/2020 11:49:44 AM

4. பின்வருவனவற்றுள் செயற்கூறு பணிமிகுப்பு சார்ந்த எந்த கூற்று சரி கிடையாது?

(அ) பணிமிகுக்கப்பட்ட செயற்கூறுகள் முன்வடிவில் வேறுபட்டு இருக்க வேண்டும்.

(ஆ) செயற்கூறு பணிமிகுப்பின் போது திருப்பி அனுப்பும் தரவினமும் கருத்தில் கொள்ள வேண்டும்.

(இ) பணிமிகுக்கப்பட்ட செயற்கூறின் முன்னியல்பு அளபுருக்கள் பணிமிகுக்கப்படும் போது கருத்தில் கொள்ளப்படுவதில்லை.

(ஈ) அழிப்பிசெயற்கூறுகள்பணிமிகுக்கப்பட முடியாது.

5. பின்வருவனவற்றுள் எது பிழையான செயற்கூறு பணிமிகுப்பு முன்வடிவாகும்?

(அ) Void fun (int x);
Void fun (char ch) ;

(ஆ) Void fun (int x);
Void fun (int y);

(இ) Void fun (double d);
Void fun (char ch);

(ஈ) Void fun (double d);
Void fun (int y);

பகுதி -ஆ

குறு வினாக்கள்

1. செயற்கூறு பணிமிகுப்பு என்றால் என்ன?
2. பணிமிகுக்க முடியாத செயற்குறிகளைப் பட்டியலிடுக.
3. class add{int x; public: add(int)}; இனக்குழுவின் வெளியே ஆக்கி வரையறுப்பை எழுதுக.
4. ஒரு செயற்கூறின் திருப்பி அனுப்பும் தரவினம் செயற்கூறு பணிமிகுப்பிற்கு உதவுமா?
5. ஒரு செயற்கூறு பணிமிகுப்பின் பயன் யாது?

பகுதி -இ

சிறு வினாக்கள்

1. செயற்கூறு பணிமிகுப்பிற்கான விதிமுறைகள் யாவை?
2. பல செயற்கூறுகள் இருக்கும் போது, நிரல் பெயர்ப்பி அவற்றுள் எந்த செயற்கூறினை செயல்படுத்த வேண்டும் என்பதை எப்படி தீர்மானிக்கும் ? எ.கா.தருக.
3. செயற்குறி பணிமிகுப்பு என்றால் என்ன? பணிமிகுப்பு செய்யக்கூடிய செயற்குறிகளுள் சிலவற்றை கூறு.
4. ஆக்கியை பணிமிகுத்தலால் விளையும் நன்மைகளைப் பற்றி விவாதிக்கவும்?
5. class sale (int cost, discount ;public: sale(sale &); குறிப்பிட்ட செயற்கூறினுக்கு ஒரு inline அல்லாத வரையறுத்தலை எழுது.

பகுதி-ஈ

பெருவினாக்கள்

1. செயற்குறி பணிமிகுப்பிற்கான விதிமுறைகள் யாவை?
2. பின்வரும் இனக்குழு நிரலைப் படித்துப் பார்த்து, (i) முதல் (v) வரையிலான வினாக்களுக்கு விடையளி

```
class Book
{
int BookCode ; char Bookname[20];float fees;
public:
Book( )                    // செயற்கூறு 1
{
fees=1000;
BookCode=1;
strcpy  (Bookname,"C++");
}
void display(float C)      // செயற்கூறு 2
{
cout<<BookCode<<":"<<Bookname<<":"<<fees<<endl;
}
~Book( )                   // செயற்கூறு 3
{
cout<<"End of Book Object"<<endl;
}
Book (intSC,char S[ ],float F)  ;   // செயற்கூறு 4
};
```

(i) மேற்கூறிய நிரலில், செயற்கூறு 1 மற்றும் செயற்கூறு 4 என்ற செயற்கூறுகளை ஒன்று சேர்த்து எவ்வாறு குறிப்பிடலாம்?

(ii) செயற்கூறு 3 எந்த கருத்துருக்களை விளக்குகிறது? இந்த செயற்கூறு எப்பொழுது அழைக்கப்படும்/ செயல்படுத்தப்படும்?

(iii) செயற்கூறு 3 பயன் யாது?

(iv) செயற்கூறு 1 மற்றும் செயற்கூறு 2 ஆகிய செயற்கூறுகளை அழைக்கும் கூற்றுகளை main () செயற்கூறில் எழுதுக.

(v) செயற்கூறு 4 க்கான வரையறையை எழுதுக .

3. பின்வரும் நிரலுக்கான வெளியீட்டை எழுதுக

```
#include<iostream>
using namespace std;
class Seminar
{
int Time;
public:
Seminar()
 {
Time=30;cout<<"Seminar starts now"<<endl;
}
void Lecture()
{
cout<<"Lectures in the seminar on"<<endl;
}
Seminar(int Duration)
{
Time=Duration;cout<<"Welcome to Seminar "<<endl;
}
Seminar(Seminar &D)
{
Time=D.Time;cout<<"Recap of Previous Seminar Content "<<endl;
}
~Seminar()
{
cout<<"Vote of thanks"<<endl;
}
};
int main()
{
Seminar s1,s2(2),s3(s2);
```

```
s1.Lecture();
return 0;
}
```

4. பின்வரும் நிரலின் அடிப்படையில் வினாக்களுக்கு விடையளி:

```
#include<iostream>
#include<string.h>
using namespace std;
class comp {
public:
char s[10];
voidgetstring(char str[10])
{
strcpy(s,str);
}
void operator==(comp);
};
void comp::operator==(comp ob)
{
if (strcmp(s,ob.s)==0)
cout<<"\nStrings are Equal";
else
cout<<"\nStrings are not Equal";
}
intmain()
{
compob, ob1;
char string1[10], string2[10];
cout<<"Enter First String:";
cin>>string1;
ob.getstring(string1);
cout<<"\nEnter Second String:";
cin>>string2;
ob1.getstring(string2);
ob==ob1;
return 0;
}
```

(i) நிரலின் இறுதி வரை நீடித்திருக்கும் பொருள்களை கூறு.

(ii) நிரலின் இயக்கத்திற்கிடையே அழிந்து விடும் பொருளை கூறு.

(iii) பணிமிகுக்கப்பட்ட செயற்குறி மற்றும் அதனை அழைக்க பயன்படும் கூற்றினை எழுதுக.

(iv) பணிமிகுப்பு செய்யப்பட்ட உறுப்பு செயற்கூறின் முன்வடிவை எழுதுக.

(v) பணிமிகுக்கப்பட்ட செயற்குறிக்கு எந்த வகையான செயலேற்பிகள் பயன்படுத்தப்பட்டுள்ளன?

(vi) எந்த ஆக்கி செயல்படுத்தப்படும்? நிரலின் வெளியீட்டை எழுது.

ஆய்வறிக்கை

தொடக்கத்தில் உன் சிறுசேமிப்பில் தொகை ரூ.500 உள்ளது என்று வைத்துக்கொள்வோம். அதில் மேலும் சில தொகையினை சேமிக்க விரும்புகிறாய். தொடக்க மதிப்பு ரூ.500 கொண்ட amount என்னும் தரவு உறுப்பைக் கொண்டு deposit என்ற இனக்குழுவை உருவாக்கு. அதில், பின்வரும் மூன்று ஆக்கிகளை வரையறுக்கவும்.

1 – அளபுருக்கள் இல்லாமல் – (எந்த தொகையும் சிறுசேமிப்பில் சேர்க்கப்பட மாட்டாது).

2 – சேமிப்பில் சேர்க்கப்பட வேண்டிய தொகையினை கொண்ட அளபுருக்கள்

3.ஒவ்வொரு முறையும் தொகையை சேர்க்கும் போது அதற்கு சமமான தொகையும் தாமாகவே சேர்த்துக் கொள்ளப்பட வேண்டும்.

Deposit இனக்குழுவிற்கு ஒரு பொருளை உருவாக்கி, சேமிப்பில் உள்ள மொத்த தொகையினை வெளியிடு.

சொற்களஞ்சியம்

முன்வடிவு	அளபுருக்களின் எண்ணிக்கை மற்றும் தரவினம்
பல்லுருவாக்கம்	பல வடிவங்கள்
முன்னியல்பு செயலுருபு	தொடக்க மதிப்பிருத்தல்

துணை நூல்கள்:

1. Object Oriented Programming with C++ (4th Edition), Dr. E. Balagurusamy, Mc.Graw Hills.
2. The Complete Reference C++ (Forth Edition), Herbert Schildt. Mc.Graw Hills.
3. Computer Science with C++ (A text book of CBSE XI and XII), SumitaArora, DhanpatRai& Co.
4. A text book of CBSE XI and XII computer science by PreetiArora and Pinky Gupta.
5. Computer Science with C++ Reeta shoo and Gagansahoo
6. The C++ Programming Language,BjarneStroustrup
7. C++ Primer (5th Edition) by S. B. Lippman, J. Lajoie