LECTURE-8

OBJECT ORIENTED PROGRAMMING C++

Dr. Farhana Sarker
Assistant Professor
CSE, ULAB

19/07/2020

CONSTRUCTOR

A **constructor** is **a special member function of a class** that is automatically called when an object of a class is created.

CHARACTERISTICS OF CONSTRUCTOR

A constructor is different from normal functions in following ways:

- Constructor has same name as the class itself
- Constructors don't have return type
- A constructor is automatically called when an object is created.
- If we do not specify a constructor, C++ compiler generates a default constructor for us (expects no parameters and has an empty body).

SYNTAX OF CONSTRUCTOR

```
class class_name
{
public:
class_name(){
Body of the constructor;
}
};
```

EXAMPLE

```
class A
{
int a;
public:
A()
{
a=10;
cout<<a;
}
};
```

```
int main()
{
A obj;
return 0;
}
```

EXAMPLE 2

```
include<iostream>
using namespace std;
class Print {
        public: int i;

    Print()
     {
      for(i=1;i<=10;i=i+1;)
          {
              if(i%2==0)
                   cout<<“Even :”<<i<<endl;
              else
                   cout<<“Odd :”<<i<<endl;
          }
  };
```

```
int main()
{ Print  T1;
return 0;

}
```

EXAMPLE 3

```
include<iostream>
using namespace std;
class Person{     string name;
                  int age;
                 public:
                 Person()
                 { cin>>name;
                   cin>>age; }
                 void output()
                 {cout<<name<<endl<<age;}
};
```

```
int main()
{ Person P1,P2,P3,P4,P5;
  cout<<"\n Output for P1 and P2:"<<endl;
          P1.output();
          P2.output(); P3.output();
          P4.output(); P5.output();
   return 0;
}
```

OUTPUT 4

Input for person1: Farhana 38

Input for person2: Selina 30

Input for person3: Parvez 23

Input for person4: Anita 40

Input for person5: Orpita 12

Output for person1: Farhana 38

Output for person:2 Selina 30

Output for person3: Parvez 23

Output for person4: Anita 40

Output for person5: Orpita 12

EXAMPLE 4

```
include<iostream>
using namespace std;
class Person{    string name;
                 int age;
                public:
                Person()
                { cin>>name;
                  cin>>age; }
                void output()
                {cout<<name<<endl<<age;}
};
int main()
{ Person obj[5]; int i;
for(i=1; i<=5; i=i+1)  {
        cout<<"Output for person" <<i<<":";
        obj[i].output(); cout<<endl;}
return 0;
}
```

TYPES OF CONSTRUCTOR

There are 3 types of Constructor:

- Default Constructor
- Parameterised Constructor
- Copy constructor

PARAMETERISED CONSTRUCTOR

```
class A
{
int a;
char b;
public:
A(){cout<<"Hello";}
A(int p){a=p; cout<<a;}
A(int i, char j)
{
a=i; b=j;
cout<<a<<b;
}
};
```

```
int main()
{
A obj1(100, 'r'), obj2(100), obj3;
return 0;
}
```

COPY CONSTRUCTOR

The **copy constructor** is a constructor which creates an object by initializing it with an object of the same class, which has been created previously.

Syntax:

class_name(class name &object name)

{ body}

EXAMPLE

```
class A
{
int a, b;
public:
A(int i)
{
a=i;
cout<<a;
}
A(A  &obj)
{
a=obj.a;
cout<<a;
}};
```

```
int main()
{
A obj1(10), obj2(obj1),
obj3(obj2);

A obj4=obj2;
return 0;
}
```

DESTRUCTOR

A destructor is a special member function that destroy (or delete) the object.

Syntax:

```
class class_name
{
public:
~class_name(){body of destructor}
};
```

RULES OF DESTRUCTOR

1) Destructors name should be same as class name.
2) Name should begin with tilde sign(~).
3) There cannot be more than one destructor in a class.
4) Destructors do not allow any parameter.
5) They do not have any return type.
6) Destructors are also automatically called.
7) When you do not specify any destructor in a class, compiler generates a default destructor and inserts it into your code.

EXAMPLE

```
#include <iostream>

using namespace std;

class HelloWorld{

public:

HelloWorld(){ cout<<”Default Constructor is called"<<endl; }

~HelloWorld(){ cout<<"Destructor is called"<<endl; }

void display(){ cout<<"Hello World!"<<endl; }

 };
```

```
int main(){
HelloWorld obj;
obj.display();
return 0; }
```

QUESTION & ANSWER

Dr. Farhana Sarker