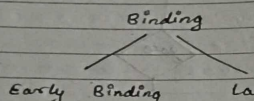
Unit - II

Binding data And functions

- Binding refers to the process of converting identifiers (such as variable and performance) into addresses.
- Binding is done for each variable and functions.
- It take place either at compile or at runtime.



→ Early Binding

As the name indicates, compiler directly associate an address to the function call.

It replaces the call with a machine language.

→ Late Binding

The compiler adds codes that identifies the kind of object at runtime then matches the call with the right function.

→ Defining a class

- class in C++ is the building block that leads to Object - Oriented programming.
- It is a user defined data type, which holds its own data member and member function.
- Ex - Consider the class of cars
- A class is defined in C++ using the keyword followed by the name of the class.
- The body of class is defined inside the curly brack and terminated by a semicolon at ends.

→ Creating an Objects

- To create an object of a given class, we specify the name of the class, followed by the name of object

Inline function

C++ provide inline function to reduce the function call overhead.

- An inline function is a function that is expanded in a line when it is called.
- When the inline function is called whole code of the inline function gets inserted.

→ Advantage

- function call overhead doesn't occur.
- Useful for embeded system.
- saves the overhead of a return call.

Constructor And Destructor

- A constructor is a member function of a class that has the same name as the class name.
  It helps to initialize the object of a class.
  It can either accept the argument or not.
  It is called whenever an instance of class is created.
  There can be many constructor in a class.

- Destructor is also a member function of a class that has the same name as the class name preceded by tilde (~) operator.
  It helps to deallocate the memory of an object.
  It is called while object of class is freed or deleted.
  It is called in reverse order of constructor.

→ Types of Constructor

• Default Constructor

If no constructor is defined in the class then the compiler automatically create one for the program.

This constructor which is created by the compiler when there is no user defined constructor and doesn't take any parameter is called default constructor.

format of default constructor

```
/*..... Format of default constructor.
class class_name
{
    .....
    Public :
        class_name ( ) { } ; // default constructor
        .....
};
```

• Parameterized Constructor

The constructor that can take arguments are called parameterized constructor.

A parameterized constructor is a constructor that contain a parametre of same class type.

• Copy Constructor

A copy constructor is a member function that initialize an object using another object of the ~~class~~ same class.

A constructor which creates an object initializing it with the object of the same class is known as copy constructor.

Copy constructor takes a refrence of the same class as an argument.

(#) Member function and Method in C++

• Member function are the function which have their declaration inside the class defination and works on the data member of the class.

• we can define member function inside the class as dot syntax (.)

• we can define member function outside the class as scope ~~resout~~ resolution (::)

→ Method are function that belong to the class which help to eliminate the need for writing the same code again and again.

- Difference between method and function

→ function is a set of instruction that perform a task.
Method is a set of instruction that are associated with an object.

→ functions are defined outside a class
Methods are defined inside a class

(#) Friend Class And Friendly Function

- A friend class can access private and protected member of other classes in which it is declared as a friend.
- We can declare a friend class by using friend keyword

Syntax

```
friend class class_name
```

```
Class CreeKs {
// GFG is a friend class of CreeKs
friend class GFG;
}
```

→ Base class

→ syntax

```
Class GFG {
Statements;
}
```

→ friend class

- A friend function is defined as a function that can access private, protected and public members of a class.
- The friend function is declared using the friend keyword inside the body of the class

Syntax

```
class className {
.....
friend return type functionName
.....
}
```

- We can declare it either in the 'public' or 'private' part
- Friend function have objects as arguments.
- It works symmetrically with all its friends.

(#) Returning objects

- A function can also return objects either by value or by reference. When object is returned by value from a function, a temporary object is created within the function, which holds the return value.
- An object can be returned by a function using the return keyword.

Page No.
Date

Syntax, Return an object

```
1 | class-name function-name (parameter-list) {
2 |     // body of the function
3 | }
```

(#) Array of an Object

An array is a collection of similar data items stored at contiguous memory location

- An Array of object stores objects
- An array of a class type is also known as an array of objects.
- The index of the array always starts with 0
- Syntax:

  Class Name Object Name [Number of Object];