Classes and Object

Manmohan
Department of Computer Science.

Abstract- **This paper is present the most essential type or features of 'C++' is 'class'. The idea of using this user defined data type i.e.'class',is all about coming by stucture used in 'c'.In which the way of creating and implement a 'class' with new and changeable features.[1]_An 'Object' is defined two things i.e., 'Fields'and 'Method'.Field means data members,characterstics and Method means the action associated with a particular Object.let we take an example,suppose we have a camel,so some of its Field might be height,weight and its speed or its Method are food,water and session.These all are considerd in C++.[2]**

I. INTRODUCTION

As we all know the unique feature of the 'c' language is structure it is user defined data type with a template that defines it's data properties.It is a tool for handling a group of logically related data item as well as packing them.for eg:

```
Struct student
{
    Char name[10];
     Int roll_number;
 };
```

From the example,the Struct keyword implement a new data type student that hold two fields of different data types but the most bigger limitation of 'Structure' is that they don't permit data hiding. It means that the members ,variables and function of the structure uses anywhere in the scope. So that's why a 'Class' is defined or implemented in 'C++'.which is used to hide the data members,variables and functions. The main difference between a 'Structure' and 'Class' in C++ is that, by default the members of a class are private,while by default the member of a Structure are Public.[1]

Some specifiers are:

Private: This specifiers is used to accessed the member of the Class only.

Protected: This specifiers is used to accessed the member of the same Class.

Public: This specifiers is used to accessed from anywhere through the object of the Class.[3]

Class implementation:

```
  Class item
{
        Private:
              Int number;
              Float cost;
        Public:
              Void getdata(int a,float b);
              Void putdata();
}
```

II. MENMBER FUNCTION

Member function are of two type:

1) Outside the class definition.
2) Inside the class definition.

III. OUTSIDE THE CLASS DEFINITION

In which the function should be declared in the class but have to define outside the class. They should have function header as well as function body.

The general form of the outside definition is:

```
Return type class-name :: function-name(argument, declaration)
{
  Function body
}
```

From the general form, class-name tells the compiler that function name belongs to class-name with the help of scope of resoultin operator(::). Moreover, member function can access the private data of the class but nonmember function can't do so. On the other hand it can call another member functon directly without using dot operator.[1]

IV. INSIDE THE CLASS DEFINITION

Another method of defining a member function is to replace the function declartion by the actual function definition inside the class.

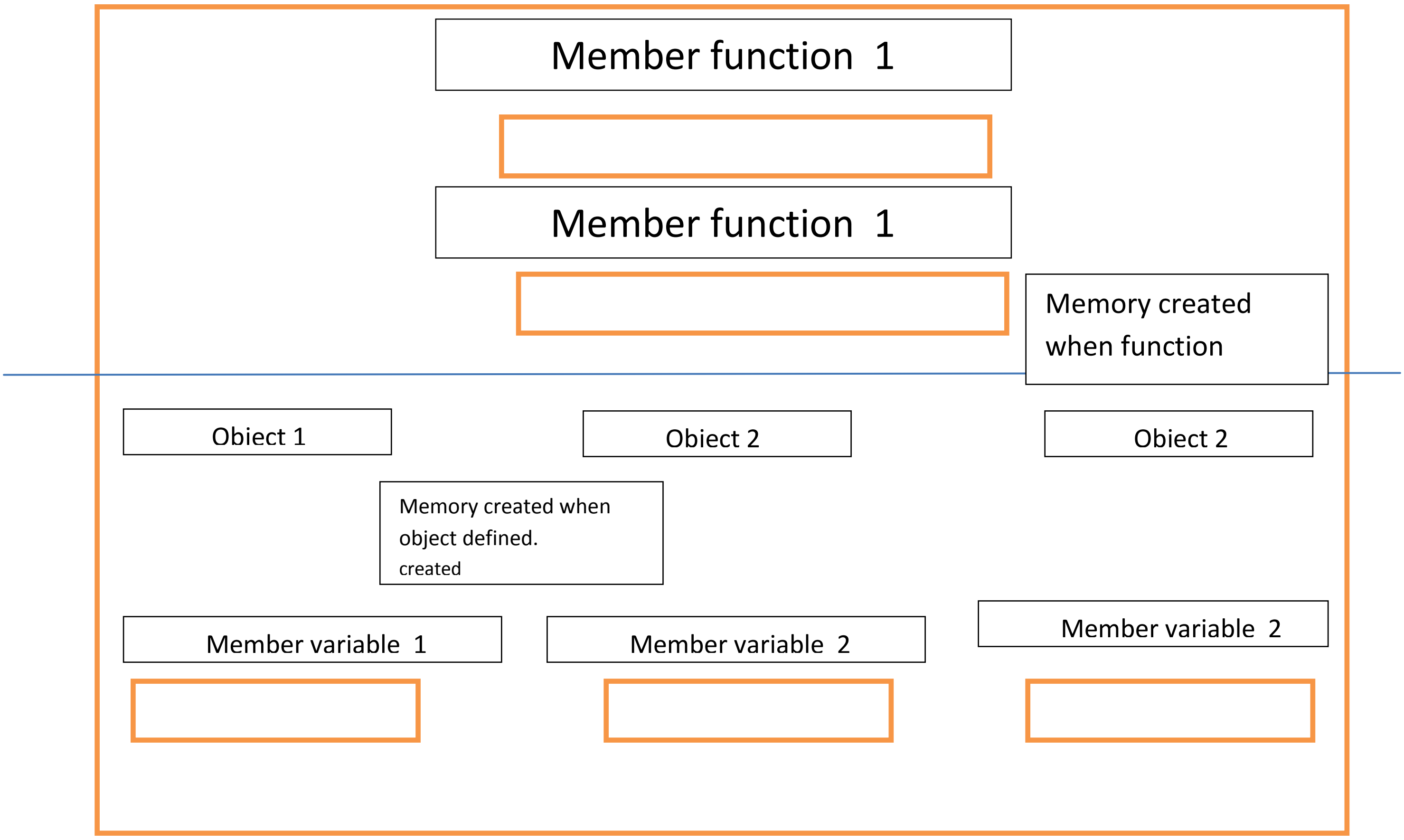
For eg.,

```
Class item
{
        Int num;
        Float cost;
  Public:
       Void getdata(int a,float b);   \\ Inline function.
       Void putdata(void)
         {
             Cout<<num<<"\n";
             Cout<<cost<<"\n";
          }
 };
```

when a function is defined inside a class it is treated as an inline function. Therefore, all the restriction and limitation are also applicable here. Normally, only small function are defined inside the class definition [1].

V. MEMORY ALLOCATION FOR OBJECT

As we all know that memory space for object is allocated when they are declared and not when the class is specified. Actually, the member function are created and placed in memory space only once they are defined as a part of a class specification. Whereas,the object are needed seprated memory allocation because member variables hold different data values and different object as given below in diagram.[1]



VI. OBJECT AS FUNCTION ARGUMENT

Like any other data type,an object may be used as a function argument. This can be done in two ways.

A) A copy of the entire object is passed to the function.

B) Only the address of the object is transferred to the function.

Our first method is also known as pass-by-value. If we have a object and it passed into function then it doesn't affect the calling function. The second method is called pass-by-reference. In which the address of the object is passed then it directly works on the actual object used in the class.[1]

VII. FRIENDLY FUNCTION

As we have discussed that the private members can't be accessed from outside the class i.e., a nonmember can't accessed to the private data of a class. To make a functon "friendly" firstly we have declare a function as given below:

```
Class ABC
{
   .........
   .........
Public:
  .........
  .........
Friend void XYZ(void);
};
```

<u>A friend function having some characterstics:</u>

1) It can be declared either in the public or in private part of the class.
2) It is not in the scope of the class to which it has been declared as friend.[1]

VIII. CONCLUSION

This paper shows that a class is an extension to the data structure data type. A class can have both variables and functions as members. By default, members of the class are private whereas that of structure are public. If a member function does not alter any data in the class, then we may declare it as a **const** member function and we use object as arguments.

REFERENCE

1) Obect oriented programming c++ by **E Balagurusamy**.

2)**https://en.m.wikibooks.org/wiki/Java_Programming/Classes,_Objects_and_Types.**

3) http://www.thegeekstuff.com/2013/02/hello-world-cpp/.