C++

What is C++

1. C++ is an object oriented programming language. which is develop by Bjarne Stroustrup at AT & T Bell Lab in 1980.

2. Initially its name was " C with class" and in 1983 The Name was changed to C++.

3. The idea of C++ comes from C increment operator (++). Therefore suggestion that C++ is an incremented version of C language It means all the features & functionality supported by C by are also supported by C++.

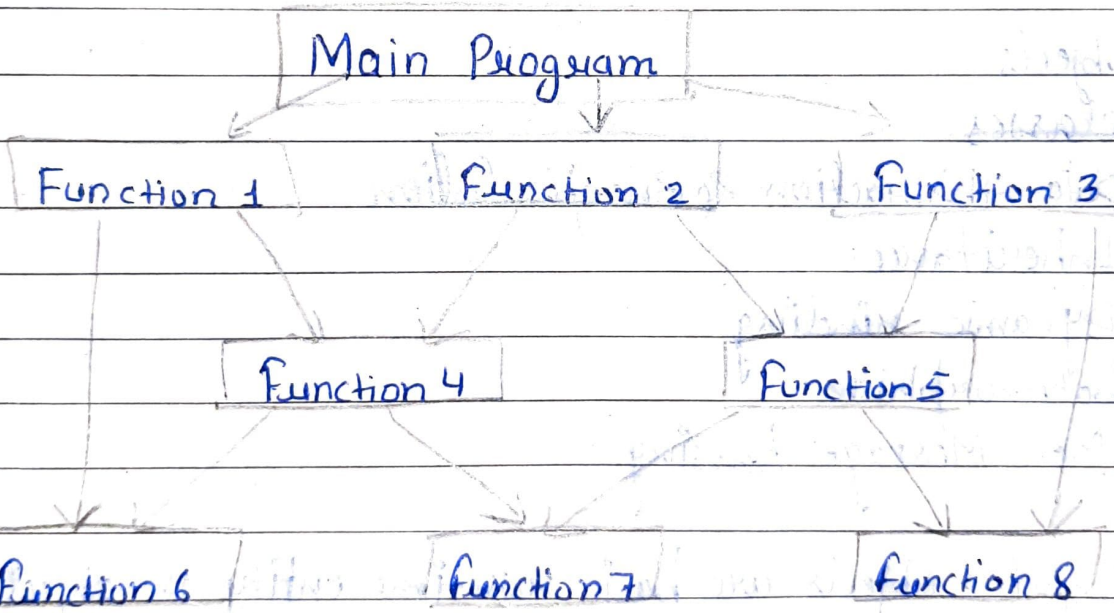
4. The most important feature that C++ add on to C language are classes, inheritance, function overloading, operator overloading which makes C++ truely object oriented programming language.

Procedure Oriented programming language

COBOL, FORTRAN, C are commonly known as procedure Oriented programming language.

COBOL :- Common Business Oriented language
FORTRAN :- Formula Translation

In procedure oriented approach the problem is viewed at sequence of things to be done such as reading, calculating, printing.



Structure of POP

Characteristic of POP structure.

* Main program is dividing into smaller program known as modules or function.
* Most of the function share global data.
* Data moves openly around the system from function

to function.

* Function transform the data from one form to other
* It follow top down approach.

Object Oriented program

It allows the composition of a program into no. of entities called object and then build data & function around the object.

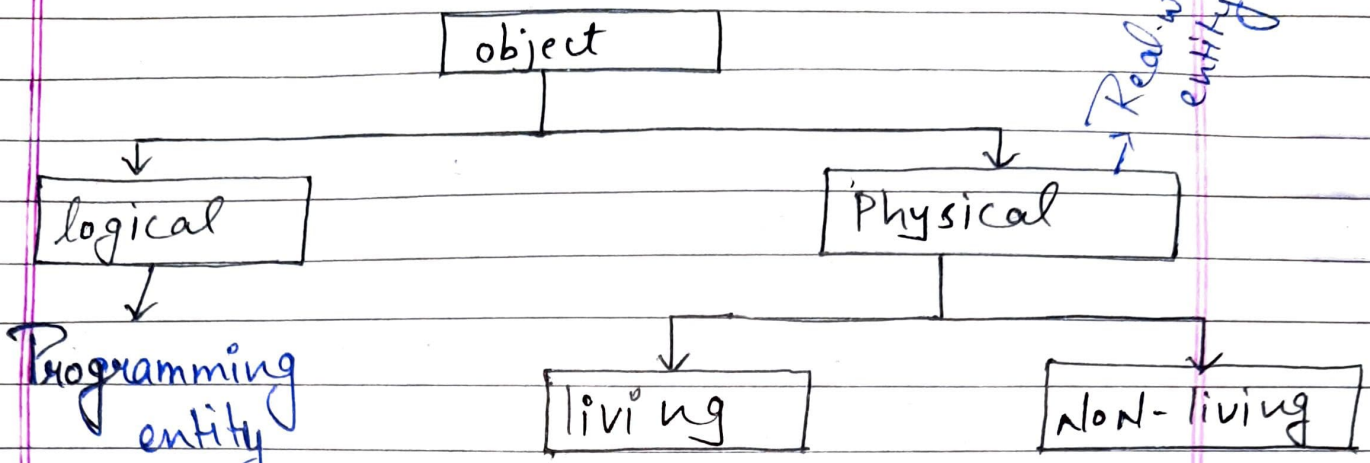
Basic concepts of object oriented programming language

1. objects
2. classes
3. Data Abstraction & Encapsulation
4. Inheritance.
5. Dynamic binding
6. polymorphism
7. Message passing

1. Object -> objects are basic runtime entity in object oriented programming language. They are used to access the elements of class. object is of class type.

"object is an instance of class".
"object is a blueprint of class".

object has the feature by which it can reflect all the properties of a class.



object = Data members + Member function.

Syntax

Declaration

Name of the class name of the object;

Eg →

student s;

Accessing member of the class :→ object can access only public members of a class.

Syntax

Calling data member & member function

Name of the object. data member;

Eg :→ s.a;

Name of the object . member function ( );

Eg → s. getdata ( );

2. Classes → class is a user defined data type. Class consist of data member & member function.
"A class is a collection of similar type of objects."
Class is a user defined data type but behaves like built in data type.

Once a class is define then any No of object can be created for that class.

Syntax

```
class   name of the class
{
  Access sepcifier:
                data member;
  Access sepcifier:
                member function;
};
```

eg:

```
class student
{
  private:
        int a;
```

```
public:
    void getdata ();
};
```

3. Data Abstraction and Encapsulation :- Wrapping up of data into a single unit is known as data Encapsulation it means the data is not accessible outside the class. Only those function which are wrapped into a class can access that data.

Data Abstraction provides the facility for representing essential information and hiding the unessential background processes.

4. Inheritance :- is the process of creating a new class from an existing class. It provides the facility of reusibility.

"Inheritance is the process by which object of one class can acquire the properties of object of another class."

Existing class is also known as base class or parent class or super class.

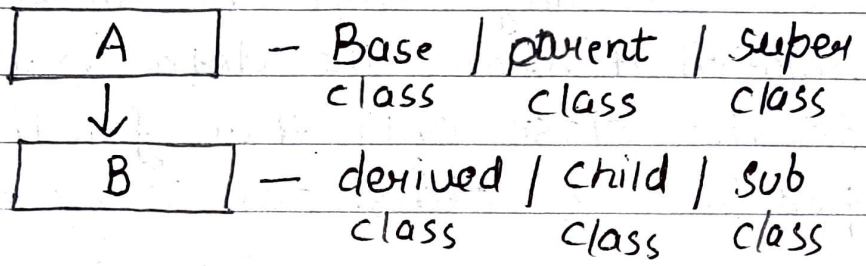
New class is also known as derived class or child class or sub class.

Types of Inheritance

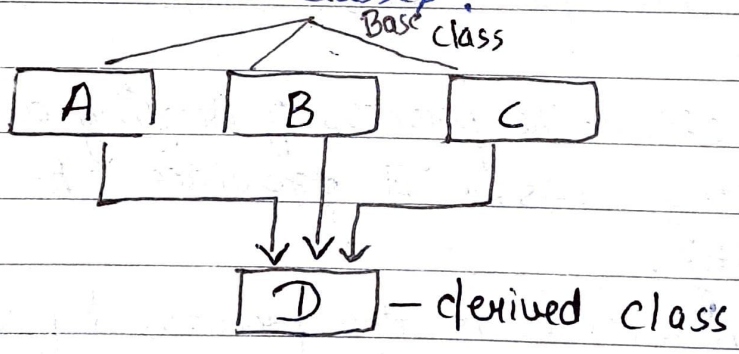
Single inheritance
Multiple Inheritance
Multi level Inheritance

Hierarical Inheritance
Hybrid ~~Inheritance~~

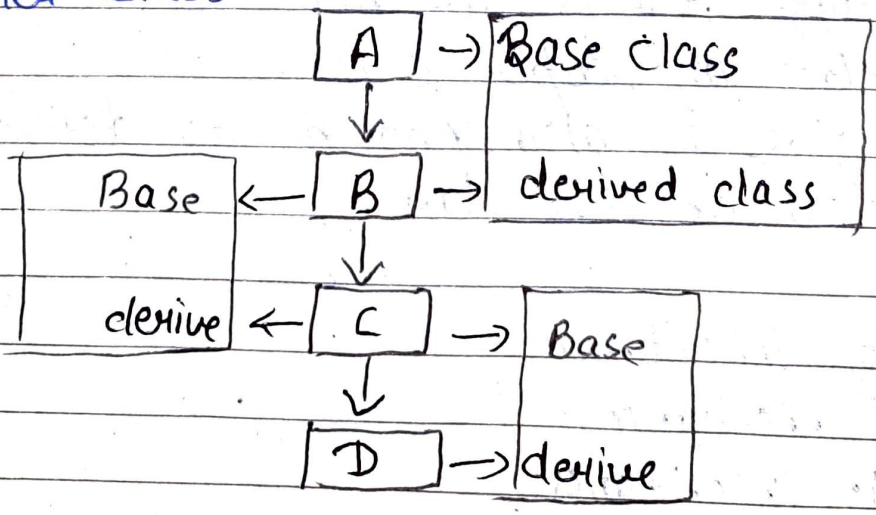
Single Inheritance :- In this a class is derived from a single class.



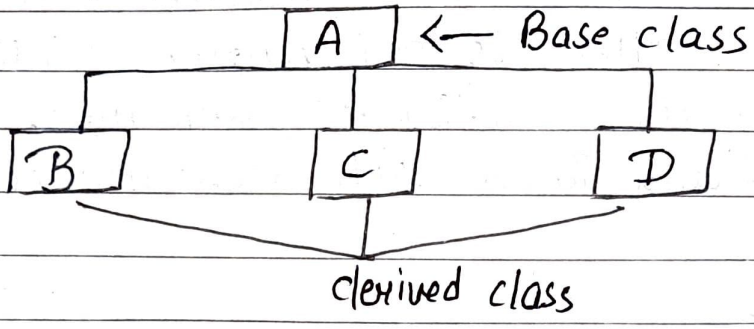
Multiple Inheritance :- In this a class is derived from more than one ~~classes~~ classes.



Multi level Inheritance :- In this one class is derived from another class



Hierarical Inheritance :- In this, one base class can have more than one derived class



Hybrid Inheritance :- It is the Combination of two or more type of inheritance.

polymorphism :- the word polymorphism is derived from two greek word poly and morphism poly means many and morphism means form so, polymorphism means anything that exist in more than one form.

" polymorphism is the ability to use an operator or function in different ways "

Dynamic Binding :- is also known as late binding or run time binding which means the code associated with a giving procedure called is not known until the time of calling the function at run time.

It is associated with inheritance and polymorphism

Message passing :- An object oriented program consist of a set of objects that communicates with each other.

Basic concepts in OOPS :-

1) Creating classes that define object and their behaviour.
2) Creating objects from class definition
3) Establishing Communication among objects with the help of functions.

"A Message for a object is a request for execution of a procedure and will invoke a function in the receiving object that generates desired result."

A Message passing involves :- Name of the Object, Name of the function [Message] and the information to send

Syntax :- Name of the object . Name of the function (Information (message))

Eg :- obj . getdata (5);

Difference btw C & C++ program

C	C++

# include <stdio.h> 	
Void main ()
{
int a;
printf ("Enter a number\n");
scanf ("%d", &a);
}
	# include <iostream.h>
	Void main()
	{
	int a;
	cout << "Enter number" << endl;
	cin >> a;
	}

① #<iostream.h> :- # is a preprocessor directive which is used for indicating the compiler that include the following statements that are present in the header file.

The Stream Keyword in the header file is used because all the data and function in the header file is in bit stream.