الذی علم بالقلم۔ علم الانسان ما لم یعلم۔

FAST

National University of Computer and Emerging Sciences

C++ (Classes and Objects)

OOP Lab-09

Computer Instructor: Engr. Khuram Shahzad

(Department Of Computer Science)

Contents

History of C++

- C++ was developed by Bjarne Stroustrup at Bell Labs in USA.
- It was initially called as “C with Classes”.
- It is super set of ‘C’ language.

- It follows bottom-up program design.
- Objects will communicate with each other.
- Objects are independent.
- It binds the data and functions together.

Procedure Oriented Programming (POP)

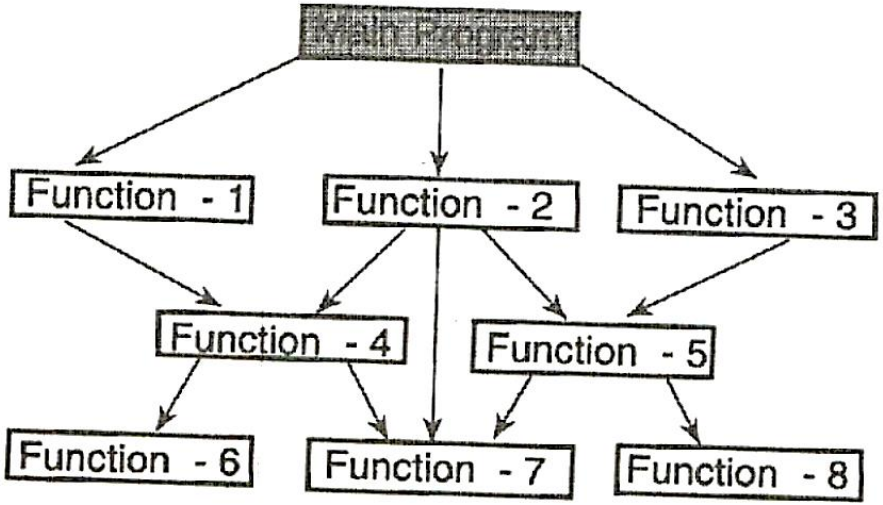


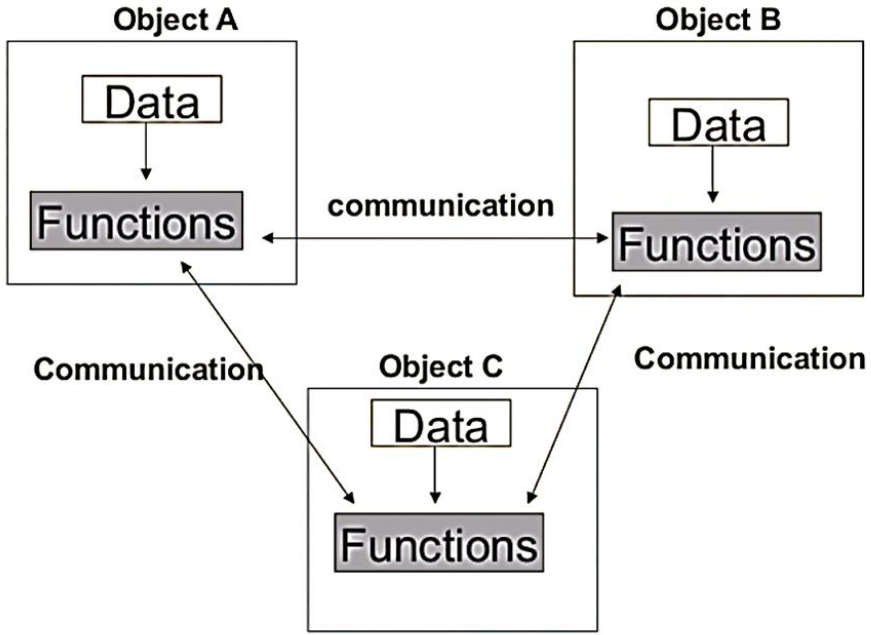
FIG 2.1 *Typical structure of procedure-oriented programs*

Limitation of POP

- Emphasis is on algorithm or procedure
- Not suitable for modeling a real world problem
- No security & integrity to the data
- Data can't be hidden
- Inheritance & Polymorphism are difficult to achieve

- Follows top down program design
- Can't reuse the existing code
- Data will be shared by many functions
- Difficult to write and understand

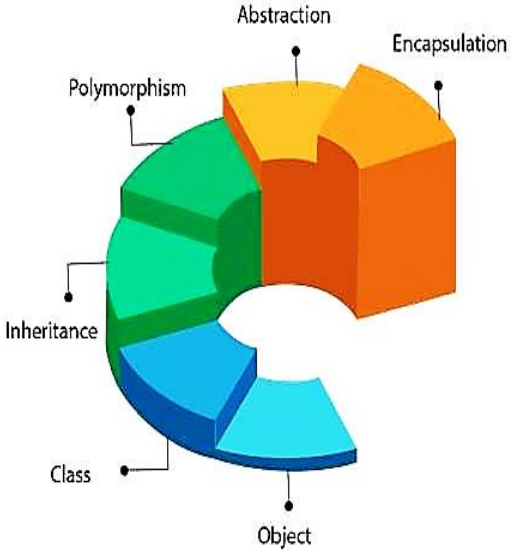
Organization Of Data & Function In OOP



Object Oriented Programming (OOP)

- The object-oriented **paradigm** is a programming methodology that promotes the efficient design and development of software systems using **reusable** components that can be quickly and safely assembled into larger systems.

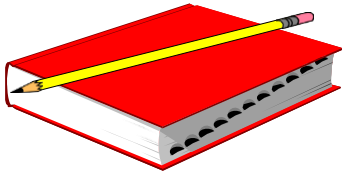
- The main aim of object-oriented programming is to implement real- world concepts like
  - Object real world entity
  - Classes Templates/ Blueprints
  - Abstraction Visibility Controls
  - Inheritance Backward Compapatibilty , parent child relation
  - Polymorphism Many forms

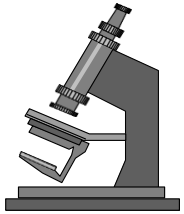
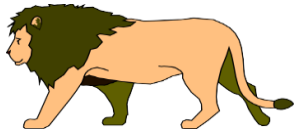


Object Oriented Programming (OOP)

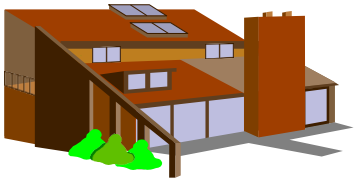
OOP tries to model the real world.
What does the real world look like?

Objects everywhere...

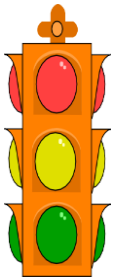


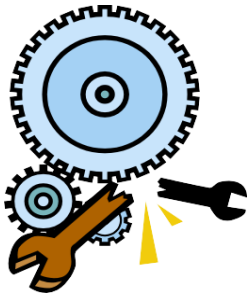
Real world entities


Objects have state...



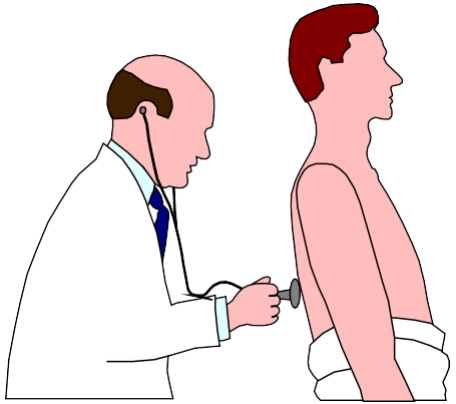
Red



Broken

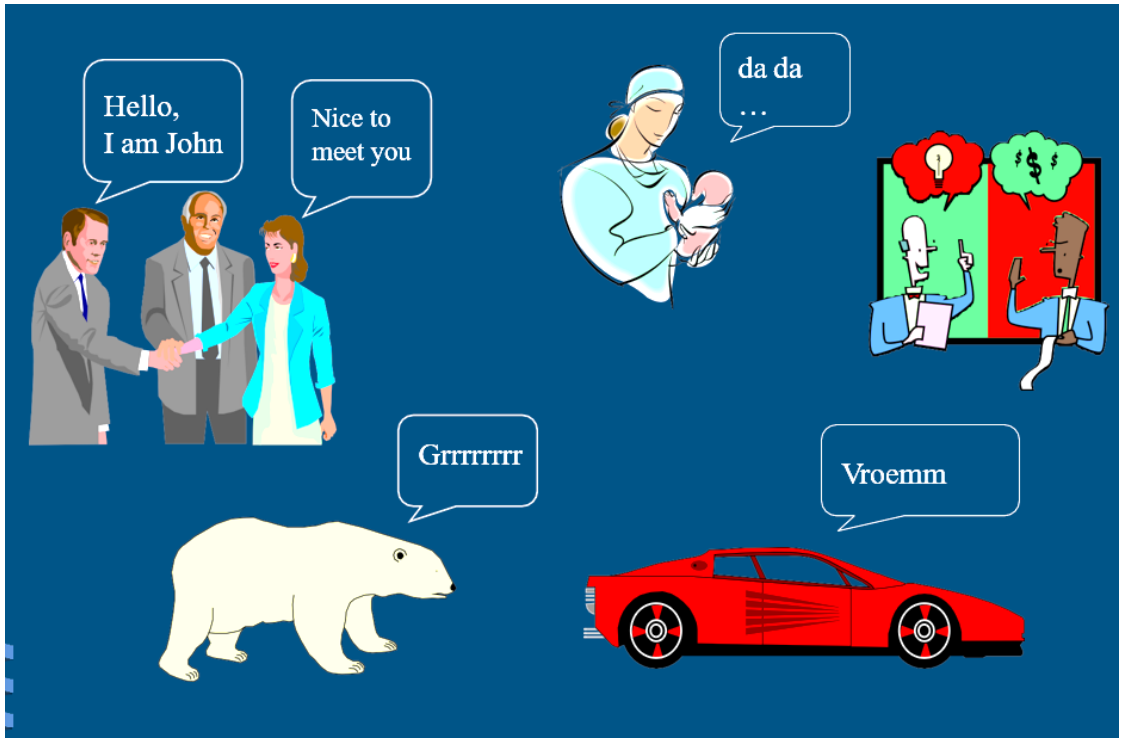
Lying

Happy



ill

Objects have behaviour….



Real World

- The world is a set of things interacting with each other.
- OOP is more natural to humans, but less natural to computers
- Computers (usually) have a single thread of control, so objects take turns

Describing the world

- **Describe a particular person**
- A man has long blond hair, green eyes, is 1.63m tall, weighs 56Kg and studies computer engineering. Now lying down asleep.

- Mahmud studies electronics, has short black hair and brown eyes.
- He is 180cm and 75 kilos. Now running to class!

- **Notice how all have specific values of**
- name, height, weight, eye color, state, …

Features Of Object Oriented Programming

- Emphasis is on data rather than the procedure
- Both data and functions are combined into a single unit
- Data can't be accesssed directly

- Data is hidden and can't be accessed by external functions
- Programs are divided into elements known as objects
- Objects may communicate with each other through functions

- Higher productivity
- Provides multiple instances of an object
- Provides security to the data
- Easy to write and understand a program

- New data and functions can be easily added
- Follows bottom up approach of program design
- We can eliminate the redundant code
- Time will be saved

Features Of Object Oriented Programming

- Data is critical element.
- Data can not be freely accessed by extenal functions
- Permits reusability of the existing code

- We can easily upgrade from small to large systems
- We can build user defined data types
- Objects are to classes as variables are to data types

Features Of Object Oriented Programming

- Artificial Intelligence & Expert systems
- Simulation & modeling
- OO databases
- Hypertext, Hyper media and Expertext
- CAD / CAM / CAE

- Decision support system
- Neural Networks
- Real time systems
- Multimedia applications
- GUI, CBTs, Office automation etc

Class

- A Class is a collection of data and functions. The data items and functions are defined within the class. Functions are written to work upon the data items and each function has a unique relationship with data items of the class.
- Classes are defined to create user defined data types. These are similar to built in data types available in all programming languages.
- Definition of data type does not create any space in the computer memory. When a variable of that data type is declared, a memory space is reserved for that variable. Similarly, when a class is defined, it does not occupy any space in the computer memory. It only defines the data items and the member function that can be used to work upon its data items. Thus defining a class only specifies its data members and the relationship between the data items through it functions.

Defining a class

A class is defined in a similar way as structure is defined. The keyword "**class**" is used to define the class. The general syntax to define a class is:

```
class ClassName
{
    Body of the class;
};
```

class is a keyword that is used to define a class.

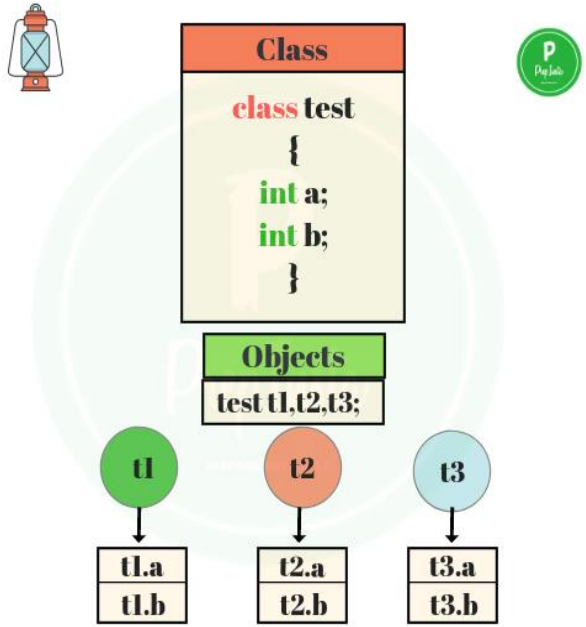
ClassName It represents the name of the class.

body of classs The body of the class consist of the data items and the functions. These are called members of the class. These are written between braces.

Semicolon (;) The body of a class ends with semicolon.

Members of a class

- A class contains data items and functions. These are called members of the class.
- The data items are called **data members** and the functions are called **member functions**.



Data Members

- The data items of a class are called data members of the class. For example a class that has four integer type and two float type data items is declared as:

In this class **int rollNo**

string name;

bool status;

float result;

```
class Student
    {
        int rollNo
        string name;
        bool status;
        float result;
    };
```

are data members of the class “**Student**”.

Member Functions

- The functions of a class that are defined to work on its data members are called member functions of the class. The member functions may be defined within the class or outside it. For example:

```
#include<iostream>
using namespace std;
class student
{
    private :
        int id;
        char name[20];
    public :
        Void Getdata(void);
        Void display (void)
        {
            cout << id <<'\t' << name << endl;
        }
};
int main( )
{
```

Data Members

Member Functions

Member Functions …

```
class Student {
    private: // private key word is an access specifire. Private
    //mean below all data member can not be access out side the class
        int a;
        int b;
        int c;
        int d;
        // public mean below all data function can be access
        //In side or out side the class. they are public
    public:         // public key word is an access specifire
        //A Member Function of Class Student
        void getData(void) {
            cout<<"Enter value of a, b and c" ;
            cin>>a>>b>>c;
        }
        //A Member Function of Class Student
        void printData(void) {
            cout<<"a= "<<a<<endl;
            cout<<"b= "<<b<<endl;
            cout<<"c= "<<c<<endl;

        }
} ;
```

In this class, there are three data members and two member functions. The member functions are "**getData**" and "**printData**". The "**getData**" function is used to input values into data members **a, b** and **c**. The "**printData**" function is used to print values of the data members on the computer screen.

Objects

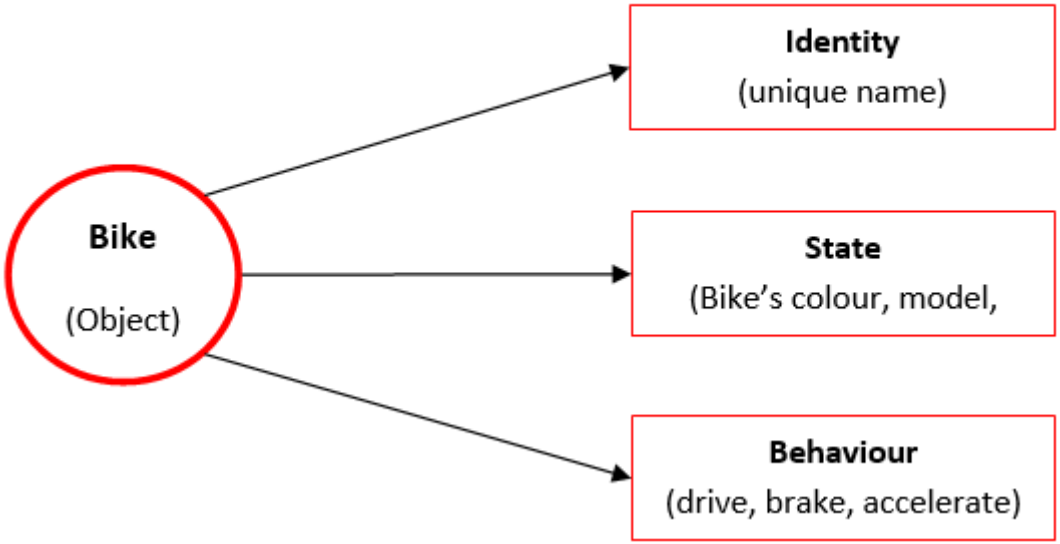
- A data type is used to declare a variable. A variable of a data type is also known as the instance or case of that data type.
- Each variable has unique name but each variable follows the rules of its data type. When a variable of a data type is declared, some space is reserved for it in the memory.
- A class is also like a data type. It is therefore used to declare variables or instances. The variables or instances of a class are called **objects**.
- A class may contain several data items and functions. Thus the object of a class consists of both the data members and member functions of the class. The combining of both the data and the functions into one unit is called data

encapsulation.

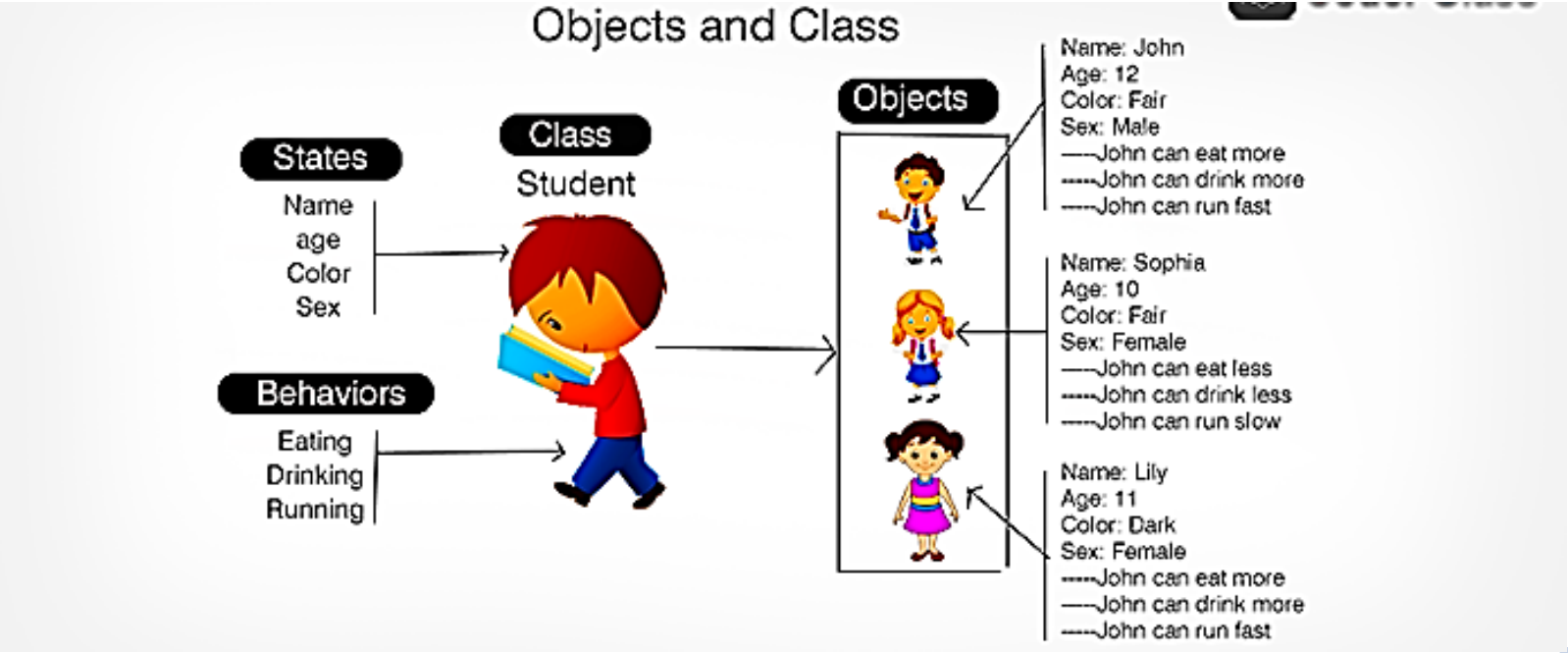
Objects…

- An object represents data members of a class in the memory. Each object of class has unique name. The name of an object differentiates it from other objects of the same class. The values of data members of different objects may be different or same. The values of data members in an object are known as the **state** of the object.
- The functions in an object are called the member functions. They are also known as the them methods. The member functions are used to process and access data of the objects.

Characteristics of Object (Identity, State & Behavior)



Characteristics Object (Identity, State & Behaviour)



References

- https://beginnersbook.com/2017/08/cpp-data-types/
- http://www.cplusplus.com/doc/tutorial/basic_io/
- https://www.w3schools.com/cpp/default.asp
- https://www.javatpoint.com/cpp-tutorial
- https://www.geeksforgeeks.org/object-oriented-programming-in-cpp/?ref=lbp
- https://www. slideshare.com

Thank You ☺