PART B

1. **Explain the basic concepts of Object oriented programming.(Nov/Dec 2011)**

- Object
- Classes
- Data abstraction
- Data encapsulation
- Inheritance
- Poiymorphism
- Dynamic binding
- Message passing

Object:

- Objects are the basic run time entities in an object oriented system.
- They may represent a person, a place, a bank account, a table of data or any item that the program has to handle.
- They may also represent user defined data such as vectors, time and lists.
- Programming problem is analyzed in terms of objects and the nature of communication between them.
- Program objects should be chosen such that they match closely with real world objects.

- When a program is executed the objects interact by sending messages to one another.
- Each object contains data and code to manipulate data.
- For example, class name object name;

Classes:

- The class is a collection of variable declarations and method definitions.
- The class contains the data and codeof an object can be made a user defined data type.
- Once a class has been defined we can create any number of objects to that class.
- Each object is associated with the data of type class which they are created.
- Class is a collection of similar objects.
- For example,

```
Class name of the class
{
    Variable declarations;
    Method definitions;
}
```

Data abstraction:

- The abstraction means provide essential things and hide non-essential things.
- Classes use the concept of abstraction and are defined as a list of abstract attributes such as size, weight, cost and functions to operate on these attributes.
- The encapsulate all the essential properties of the objects that are to be created.
- The classes use the concept of data abstraction, they are known as abstract data types.

Data encapsulation:

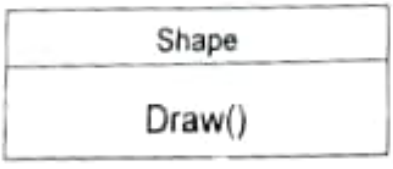
- The wrapping up of data and functions into single unit is called as **encapsulation**.
- Data encapsulation is the most striking feature of a class.
- These functions provide the inheritance between objects data and the program.
- The insulation of the data from direct access by the program is called data hiding.
- They encapsulate all the essential properties of the objects that are to be created.

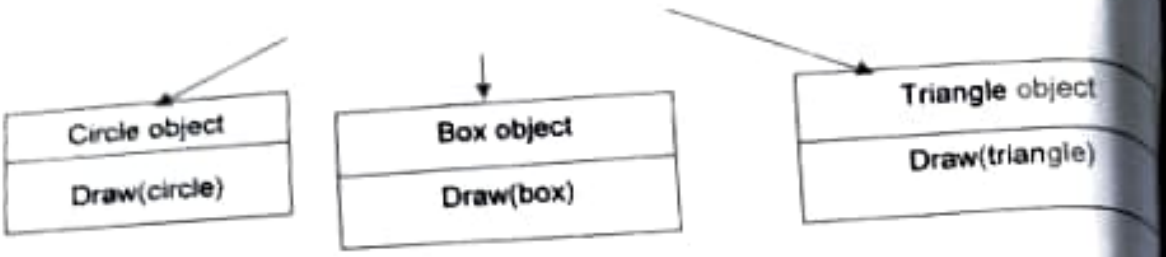
Inheritance:

- The inheritance means derived a new class from the existing one.
- It is the process by which objects of one class acquire the properties of objects of another class.
- It supports the concepts of hierarchical classification.
- The concept of inheritance provides the idea of reusability.
- The new class will have the combined features of both the classes.

Polymorphism:

- The polymorphism means the ability to take more than one form.
- An operation may exhibit different behaviors in different instances. The behavior depends upon the types of data used in the operation.
- The process of making an operator to exhibit different behaviors in different instances is known as operator overloading.
- A single function name can be used to handle different number and different types of arguments.
- Using a single function name to perform different types of tasks is known as function overloading.
- The polymorphism is extensively used in implementing inheritance.





Dynamic binding:

- Dynamic binding means the code associated with a given procedure call is not known until the time of the call at run time.
- It is associated with polymorphism and inheritance.
- Binding consist of two types. They are dynamic binding and static binding.
- The static binding means compile time process and the dynamic binding means run time process.

Message passing:

- As object oriented program consists of a set of objects that communicate with each other.
- The process of programming in an object oriented language, therefore involves the following basic types.
- Creating classes that define objects and their behavior.
- Creating objects from class definitions
- Establish communication among objects.
- A message of an object is a request for execution of a procedure and therefore will invoke a function in the receiving object that generates the desired result.

The advantage of Object Oriented methodology

- The principle of data hiding helps the programmer to build secure progra that cannot be invaded by code in other parts of the program.
- It is possible to have multiple instances of an object to co-exist without an interfaces.
- Object oriented systems can be easily upgraded from small to large syst

- Software complexity can be easily manages.
- It is possible to map objects in the problem domain to those in the program.
- It is easy to partition the work in a project based on objects.
- The data centered design approach enables us to capture more details of a model in implementable form.
- Message passing techniques for communication between objects makes the interface descriptions with external systems much simpler.
- Through inheritance, we can eliminate redundant code and extend the use of existing classes.
- It is possible to save development and their higher productivity.