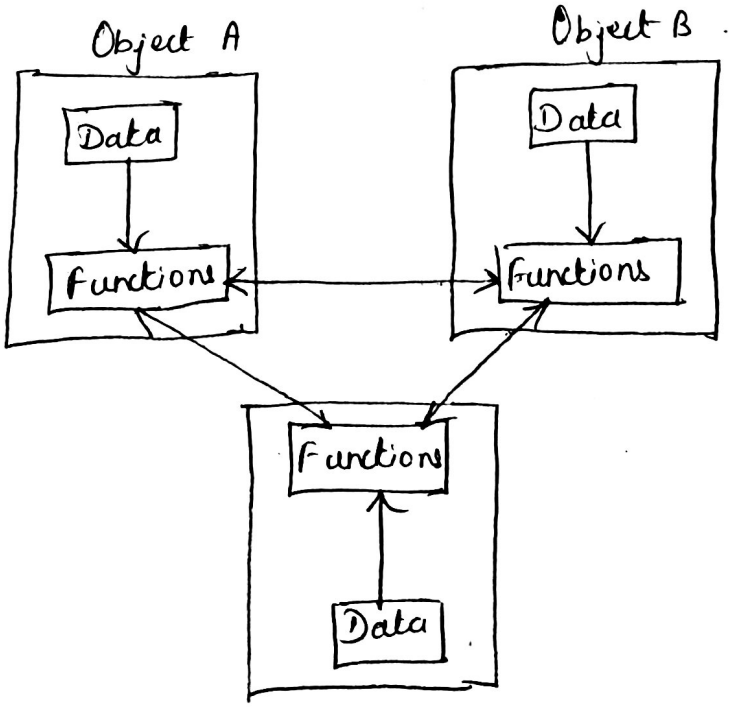
Object Oriented Programming paradigm:

- Programs are divided into number of entities called objects.
- The data of an object can be accessed only by the functions associated with that object. Thus data is hidden for external functions.
- Data structures are designed such that they characterize the objects.
- Emphasis is on data rather than procedure.
- Objects can communicate with each other through functions.
- Follows bottom-up approach in program design.



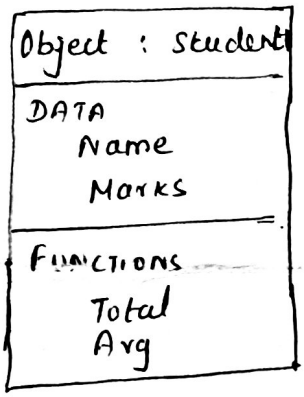
Fig(iv) Organisation of data and function in OOP

* Basic elements of OOP

(i) Objects

- Objects are basic run-time entities of OOP.
- They may represent a person, a place, a bank account or any item the program can handle and should match with real-world objects.
- Objects take up space in memory and have an associated address like structure in C.
- When a program is executed, the objects interact by sending messages to each other.

(Eg)



(ii) Class

- Collection of similar objects.
- Template that defines characteristics of object
- (Eg) Mango, apple and orange are members of the class fruit.

* Characteristics of OOP languages

(i) Encapsulation :

- The wrapping up of data and functions into a single unit (called class) is known as encapsulation.
- Data is not accessible to outside world & only the wrapped function can access it.
- This insulation of data from direct access by the program is called data hiding

(ii) Data Abstraction :

- Abstraction is the act of representing essential features without including background details.
- Classes use the concept of abstraction to encapsulate all essential properties of objects to be created such as its attributes and fns.
- Thus classes are known as Abstract Data Types (ADT)

(iii) Inheritance :

- Inheritance is the process by which objects of one class acquire the properties of objects of another class.
- Allows the extension and reuse of existing code without having to rewrite the code from scratch.
- Involves creation of new class (derived class) from the existing one (base class).
- Derived class inherits the members of base class and also adds its own.
- Inheritance can be
  - ⇒ Single inheritance - deriving a class from single base class - supported by C++

→ Multiple inheritance - deriving a class from more than one base class - supported by C++

(iv) Polymorphism

- Polymorphism means the ability to take more than one form
- in OOP, a single name / operator can be associated with different operations depending on the type of data used.

(Eg) Consider addition operation

- for two nos. a sum will be generated.
- for two strings, a concatenated string is generated.

- In C++, it is achieved by function overloading, operator overloading and dynamic binding (The code associated with a given procedure call is not known until the time of call at run time)

(v) Message passing

- It is the process of invoking an operation on an object.
- In response to the message, the corresponding function is executed in the object.

(vi) Extensibility:

- allows extension of the functionality of the existing software components.
- In C++, it is achieved through abstract classes and inheritance.

(vii) Persistence:

- the phenomenon where the object (data) outlives the program execution time and exists b/w execution of a program.
- Supported by database system.
- Not supported in C++. But can be built explicitely using file streams