Assignment - 22

Q.1 what is meant by runtime polymorphism ?

→ ○ Polymorphism is the sigle name & multiple behaviours.

○ Runtime polymorphism : The concept of Runtime polymorphism called as late binding. The decision taken here is in the late / runtime phase.

○ To achieve runtime polymorphism we ~~to~~ use the concept of overriding i.e function overriding.
- virtual & pure - virtual.

○ The runtime polymorphism means overriding the functions at runtime.

Q.2 what is the difference between overloading and overriding ?

→ 1. Overloading is the concept from compile time polymorphism.

overriding is the concept from runtime polymorphism.

2. • overloading means using the same name function or operator to perform multiple tasks.

• Overriding means using same named functions with the different defination in different classes.

3. • overloading can be done of function & operator.

• overriding can be done of only function.

4. • In overloading the function we keep the name of function same but with different prototype.

• In overriding we use 2 concepts i.e

1. virtual
2. Pure pure virtual

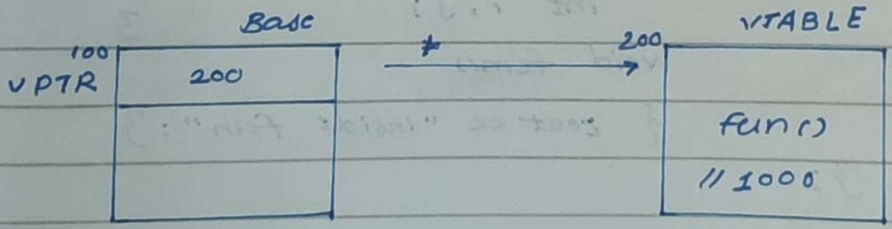
we make functions of differt definations in different classes.

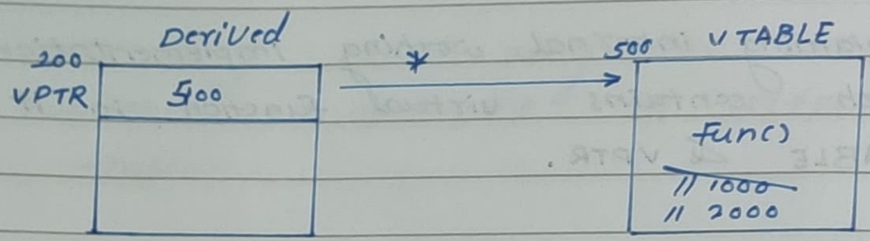
Q.3 Explaining internal working implementation of class which contains virtual function in it with V TABLE & VPTR.

-> Example -

```
          class  Base
   {
          public :
          virtual void fun()      //1000
          { }
   };
   class  Derived  :  public  Base
   {
          public:
             void fun ()        // 2000
             { }
   };
   int  main ()
 {
      Base * Bp  =  new Derived ;
          Bp -> fun () ;
 }
```

-> • In the above example Base class contains one virtual function therefore the object of Base class will have 4 bytes for the VPTR which will point to the ~~VPTR~~ VTABLE.





Q.4 What is meant by upcasting and downcasting & explain with object oriented example.

→ • In simple words upcasting means low / small capacity pointer points to high / large data.

• Downcasting high / Large capacity pointer points to small capacity data.

eg. →

```
class Base
{   public:
        int i;
    void fun() {}
      virtual void gun() {}
};
class Derived : public Base
{   public :
        int i, j;
    void fun()
    { cout << "Inside fun"; }
};
```

```
int main()
{
Base * BP = new Base;
// No casting
Base * BP1 = new Derived
// Upcasting
Derived * DP = new Base;
// Down casting
}
```

Q.5 If the base class & derived class has same named & prototyped method then which is as overloading or overriding or redefination?

- It will be called as method overriding. because in overloading same named function with diffent pstotype and in case of redefination functions having different definations.

Q.6 In which scenario of class first 4 bytes are reserved as VPTR in objects layout.

- If the class having function with virtual keyword in this case first 4 bytes are reserved for VPTR in objects layout.

Q.7 what are necessary things in our application which is used to achieve runtime polymorphism

→ following are the things necessary for overriding:

1. Alcast single level inheritance in function
2. Redefination of the function
3. Upcasting in code.

Q.8 can we override private virtual from base class into derived class ?

→ • we cannot override virtual private function from Base class because it cannot be accessed outside the class.

Q.9 why the concept of downcasting is not allowed?

→ • The concept of downcasting its not allowed because. the compiler does not allow large capacity pointer to points to the small data.

Q.10 can we define static virtual function in a class ?

→ • we can declared the static function but when we call that function we need class name & scope resolution operator without object creation.

• for this reason cannot make static virtual function.