1. Create a class account that stores customer name, account number and type of account. From this derive the classes currentAccount and savingsAccount to make them more specific to their requirements. Include necessary member functions in order to achieve the following tasks:

    1. Accept deposit from a customer and update the balance

    2. Display the balance

    3. Compute and deposit the interest

    4. Permit withdrawal and update the balance

    5. Check the minimum balance, impose necessary penalty and update the balance.

2. Write a program that uses a function *account*. The program will ask the user to enter a balance and credit amount, which it will pass to the function *account*. The function *Account* will receive the balance and credit. It will then ask the user to enter a deposit. The function will add the deposit to the credit and update the balance (balance plus credit). It will return to the calling program (main()) the balance and credit. The balance and credit will be displayed in the main().

3. What is output of the following program?

```
class student
{
   public : int marks;
         void disp()
         {
                 cout<<"its base class"
         };
         class topper:public student
         {
                 public :
                 void disp()
                 {
                         cout<<"Its derived class";
                 }
         }
         void main() { student s; topper t;
         s.disp();
         t.disp();
}
```

4. Which class/set of classes can illustrate polymorphism in the following code:

```
abstract class student
{
  public : int marks;
  calc_grade();
}
class topper:public student
{
   public : calc_grade()
   {
      return 10;
   }
};
class average:public student
{
   public : calc_grade()
   {
      return 20;
   }
};
class failed{ int marks; };
```

5. In multiple inheritance, if class C inherits two classes A and B as follows, which class constructor will be called first:

```
class A{ };
class B{ };
class C: public A, public B{  };
```