Problem statement:

Create one **C++** application which implements all type of inheritance.

Solution:

```
//The following program implements all types of inheritance.
//Author : Hetav Trivedi

#include<iostream>
#include<string>
using namespace std;
////////////////////////////

//Defining a class named 'executive' with one public function that can get &
change the value of passcode.
class executive
{

//make passcode protected
        protected:
        int passcode; //variable to get the value of passcode
        public:
        executive() // function that would allow to change the passcode
    {
    }
        executive(int p): passcode(p)
    {
    }
};

////////////////////////////

class employee
{
```

```
    protected:
    int eno; // employee number
    public:
    string name;
    float sal; // salary
    employee() // function to get the values
        {
        }
            employee(int e,string s): eno(e), name(s), sal(0)
        {
        }
        virtual void disp() //Display output.
            {
            cout<<"E Number: "<<eno
            <<"\nName: "<<name
            <<"\nSalary: "<<sal<<endl;
            }
};

//////////////////////////

//inheritance

class manager : public employee, private executive
{
    private:
    string dept;
    public:
    manager()
    {
    }
    manager(int e,string s,string d,int p=9998) :
    employee(e,s), dept(d), executive(p)
    {
    }
    int getpass(){
    return passcode;
}
void set_sal(employee& e)
    {
    int pass;

    cout<<"You are attempting to change "
    <<e.name<<"'s Salary.\n Enter your manager passcode: ";

    //passcode is 1008. Otherwise it won't work,
    //and would give the output shown below.
```

```
    cin>>pass;
        if(pass!=getpass())
            {
            cout<<"INCORRECT PASSWORD! Exiting Function."<<endl;
            return;
            }

        cout<<"Last updated Salary of employee: "
        <<e.sal<<endl;

        float s;

        cout<<"Enter new Salary: ";
        cin>>s;

        //updating data

        e.sal=s;

        cout<<"Salary updated.\n"<<endl;
        cout<<e.name<<"\'s details: \n";
        e.disp();
    }
};

////////////////////////////

class tech : public employee
{
    protected:
    int exp;
    public:
    tech()
    {
    }
    tech(int e,string s,int ex): employee(e,s),exp(ex)
    {
    }
};

////////////////////////////

class repair: public tech
{
    private:
    float costs;
    public:
    repair()
```

```
    {
    }
    repair(int e,string s,int exp) : tech(e,s,exp), costs(0)
    {
    }

//getcosts function will show the repairing cost

void getcosts()
    {
    float c;
    cout<<"Enter Repair cost: ";
    cin>>c;

    costs+=c;
    cout<<name<<" has recorded Rs."<<c<<endl
    <<"Total costs: Rs."<<costs<<endl<<endl;
    }

//Dispay all the data

void disp()
    {
    cout<<"E Number: "<<eno
    <<"\nName: "<<name
    <<"\nSalary: "<<sal
    <<"\nExperience: "<<exp
    <<"\nCosts of repairs: "<<costs<<endl;
    }
};

////////////////////////////

//initializing the main function

int main()
{
    manager claire(1005,"Claire","Tech",1008);
    employee aaron(2645,"Aaron");
    repair doug(4056,"Doug",3);

    claire.set_sal(aaron); //set the salary
    cout<<endl;

    claire.set_sal(doug); // doug's salary is changed now
    cout<<endl;

    doug.getcosts(); //
```

```
    doug.disp();
    cout<<endl;

    doug.getcosts();
    doug.disp();
    cout<<endl;

    claire.set_sal(aaron);
    cout<<"\n\n Author: Hetav. "<<endl;

    return 0;
}
```