INHERITENCE

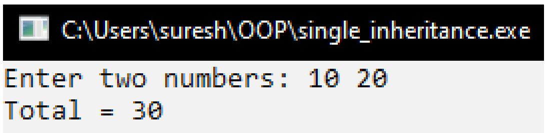
1. **Write a C++ program to add two numbers using single inheritance. Accept these two numbers from the user in base class and display the sum of these two numbers in derived class.**

PROGRAM

```
#include<iostream>
using namespace std;

class Base {
        protected:
                int x, y;
        public:
                void get_data() {
                        cout<<"Enter two numbers: ";
                        cin>>x>>y;
                }
};

class Derived: protected Base {
        private:
                int total;
        public:
                void display_total() {
                        get_data();
                        cout<<"Total = "<<x+y;
                }
};
```

```
int main() {
        Derived d;
        d.display_total();

        return 0;
}
```

OUTPUT



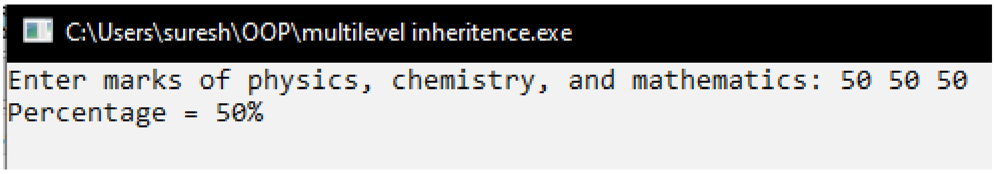
```
C:\Users\suresh\OOP\single_inheritance.exe
Enter two numbers: 10 20
Total = 30
```

2. **Write a C++ program to calculate the percentage of a student using multi-level inheritance. Accept the marks of three subjects in base class. A class will be derived from the above-mentioned class which includes a function to find the total marks obtained and another class derived from this class which calculates and displays the percentage of student.**

PROGRAM

```
#include <iostream>
using namespace std;
class Marks {
        protected:
                float p, c, m;
        public:
                void get_data() {
                        cout << "Enter marks of physics, chemistry, and mathematics: ";
                        cin >> p >> c >> m;
                }
}
class Total : protected Marks {
        protected:
                float total;
        public:
                void calculate_total() {
                        get_data();
                        total = p + c + m;
                }
};

class Percent : private Total {
        private:
                float percentage;
        public:
                void display() {
                        calculate_total();
                        percentage = total / 300 * 100;

                        cout << "Percentage = " << percentage << "%" << endl;
                }
};

int main() {
        Percent p;
        p.display();

        return 0;
}
```

OUTPUT



```
Enter marks of physics, chemistry, and mathematics: 50 50 50
Percentage = 50%
```

3. **Write a C++ program to design a base class Person(name, address, phone). Derive a class Employee(eno, ename) from Person. Derive a class Manager(designation, dept_name, basic_salary) from Employee. Write a program to:**
    a. **Accept all details of 'n' managers**
    b. **Display manager having highest salary**

PROGRAM

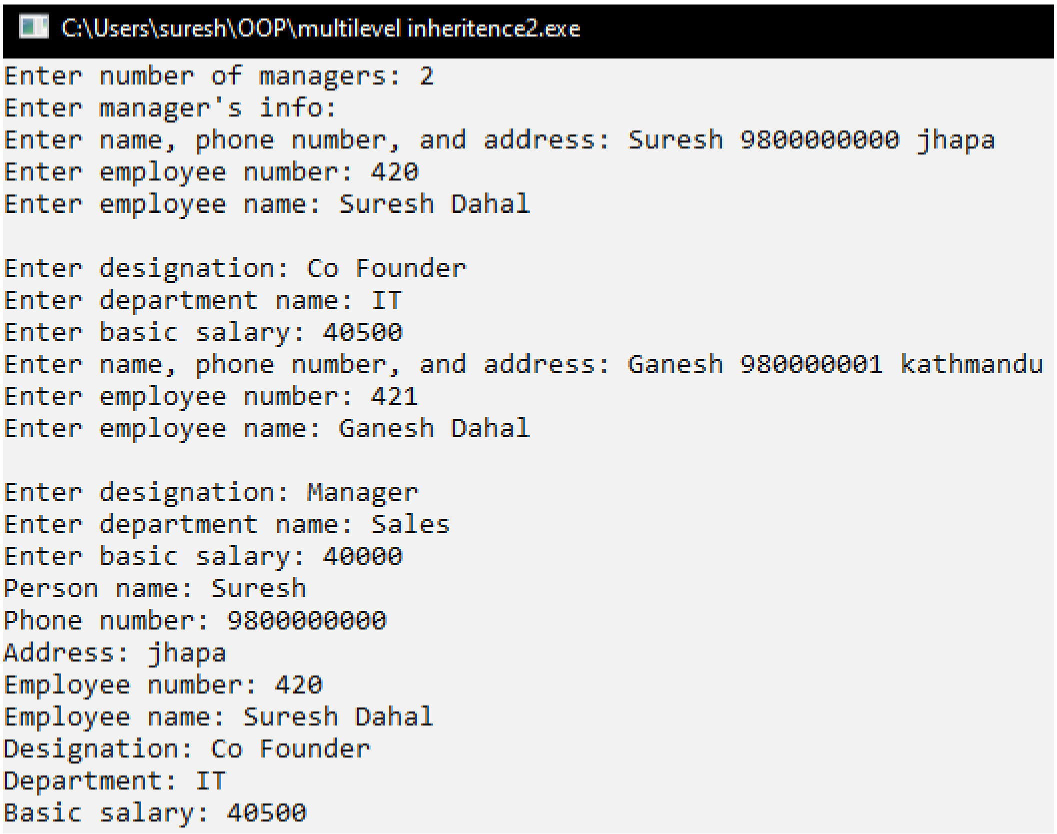
```
#include<iostream>

#include<cstring>

using namespace std;


class Person {

        protected:

                string name, phone, address;

        public:

                void get_person_info() {

                        cout<<"Enter name, phone number, and address: ";

                        cin>>name>>phone>>address;

                }


                void display_personal_info() {

                        cout<<"Person name: "<<name<<endl;

                        cout<<"Phone number: "<<phone<<endl;

                        cout<<"Address: "<<address<<endl;

                }

};


class Employee: protected Person {

        protected:

                int eno;

                string ename;
```

```
	public:
		void get_emp_info() {
			cout<<"Enter employee number: ";
			cin>>eno;
			cin.ignore();
			cout<<"Enter employee name: ";
			getline(cin, ename);
		}

		void display_emp_info() {
			cout<<"Employee number: "<<eno<<endl;
			cout<<"Employee name: "<<ename<<endl;
		}
};

class Manager: protected Employee {
	protected:
		string designation, dept_name;
		float basic_salary;
	public:
		void get_manager_info() {
			get_person_info();
			get_emp_info();
			cin.ignore();
			cout<<"Enter designation: ";
			getline(cin, designation);
			cout<<"Enter department name: ";
			cin>>dept_name;
			cout<<"Enter basic salary: ";
			cin>>basic_salary;
```

```
                }

                float get_salary() {
                        return basic_salary;
                }

                void display_manager_info() {
                        display_personal_info();
                        display_emp_info();
                        cout<<"Designation: "<<designation<<endl;
                        cout<<"Department: "<<dept_name<<endl;
                        cout<<"Basic salary: "<<basic_salary;
                }
};

int main() {
        int n;
        cout<<"Enter number of managers: ";
        cin>>n;

        Manager m[n];

        cout<<"Enter manager's info: "<<endl;
        for(int i=0; i<n; i++) {
                m[i].get_manager_info();
        }

        int index_of_manager_having_max_salary = 0;
        for(int i=0; i<n; i++) {
                if(m[index_of_manager_having_max_salary].get_salary() < m[i].get_salary())
{
```

```
                        index_of_manager_having_max_salary = i;
                }
        }
        cout<<"Manager with highest salary: "<<endl;
        m[index_of_manager_having_max_salary].display_manager_info();


        return 0;
}
```

OUTPUT



C:\Users\suresh\OOP\multilevel inheritence2.exe

```
Enter number of managers: 2
Enter manager's info:
Enter name, phone number, and address: Suresh 9800000000 jhapa
Enter employee number: 420
Enter employee name: Suresh Dahal

Enter designation: Co Founder
Enter department name: IT
Enter basic salary: 40500
Enter name, phone number, and address: Ganesh 980000001 kathmandu
Enter employee number: 421
Enter employee name: Ganesh Dahal

Enter designation: Manager
Enter department name: Sales
Enter basic salary: 40000
Person name: Suresh
Phone number: 9800000000
Address: jhapa
Employee number: 420
Employee name: Suresh Dahal
Designation: Co Founder
Department: IT
Basic salary: 40500
```

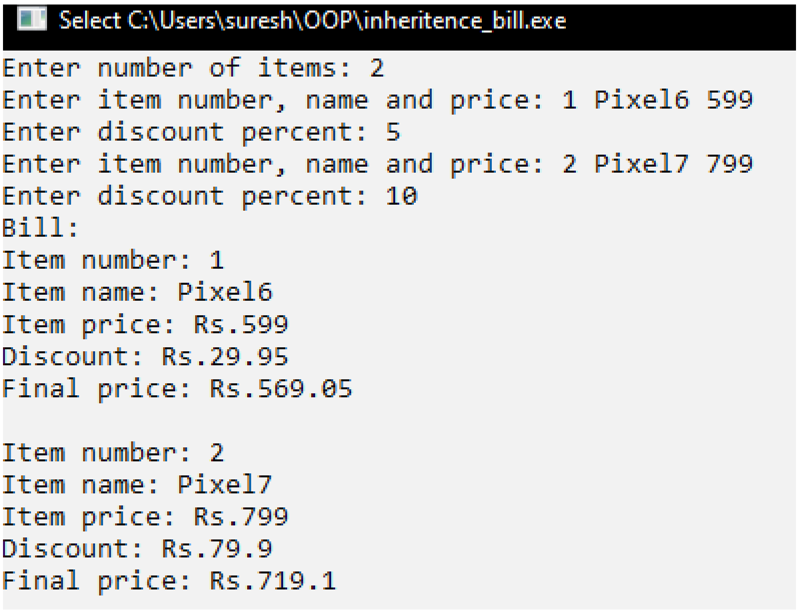
```
}
```

4. **Write a C++ program to define a base class Item (item-no, name, price). Derive a class Discounted-Item (discount-percent). A customer purchases 'n' items. Display the item-wise bill and total amount using appropriate format.**

PROGRAM

```
#include<iostream>
using namespace std;

class Item {
        protected:
                int item_no;
                string name;
                float price;
        public:
                void get_item_info() {
                        cout<<"Enter item number, name and price: ";
                        cin>>item_no>>name>>price;
                }
};

class DiscountedPrice: public Item {
        protected:
                float discount_percent;
        public:
                void get_data() {
                        get_item_info();
                        cout<<"Enter discount percent: ";
                        cin>>discount_percent;
                }
        void display_bill() {
            float discount_price = (price * discount_percent) / 100;
            cout<<"Item number: "<<item_no<<endl;
            cout<<"Item name: "<<name<<endl;
            cout<<"Item price: Rs."<<price<<endl;
            cout<<"Discount: Rs."<<discount_price<<endl;
            cout<<"Final price: Rs."<<price - discount_price<<endl<<endl;
        }
};

int main() {
    int n;
    cout<<"Enter number of items: ";
    cin>>n;

    DiscountedPrice I[n];

    for(int i=0; i<n; i++) {
        I[i].get_data();
    }

    cout<<"Bill: "<<endl;
    for(int i=0; i<n; i++) {
        I[i].display_bill();
    }

    return 0;
}
```

OUTPUT



```
Enter number of items: 2
Enter item number, name and price: 1 Pixel6 599
Enter discount percent: 5
Enter item number, name and price: 2 Pixel7 799
Enter discount percent: 10
Bill:
Item number: 1
Item name: Pixel6
Item price: Rs.599
Discount: Rs.29.95
Final price: Rs.569.05

Item number: 2
Item name: Pixel7
Item price: Rs.799
Discount: Rs.79.9
Final price: Rs.719.1
```

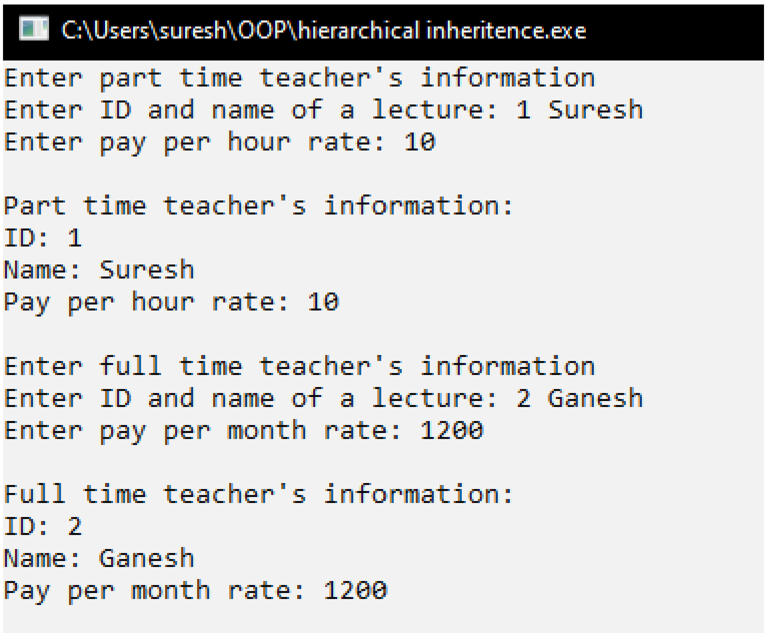
5. **Imagine in a college hires some lectures. Some lectures are paid in period basic, while others are paid in month basic. Create a class called lecture that stores ID and name of lectures. From this class derive two classes: part time, which adds payperhr (type float); and full time, which adds pay per month (type float). Each of these three classes should have a readdata () function to gate its data from user at the key board and printdata () function to display the data. Write a main ()program to test the Full time and part time classes by creating instance of them, asking the user to fill third data with readdate () and display the data with printdata().**

PROGRAM

```
#include<iostream>

using namespace std;


class Lecture {

        protected:

                int id;

                string name;
        public:
                void read_data() {
                        cout<<"Enter ID and name of a lecture: ";
                        cin>>id>>name;
                }
                void print_data() {
                        cout<<"ID: "<<id<<endl;
                        cout<<"Name: "<<name<<endl;
                }
};
class PartTime: public Lecture {
        protected:
                float payperhr;
        public:
                void read_data() {
                        Lecture::read_data();
                        cout<<"Enter pay per hour rate: ";
                        cin>>payperhr;
                }
                void print_data() {
                        Lecture::print_data();
                        cout<<"Pay per hour rate: "<<payperhr<<endl;
                }
};

class FullTime: public Lecture {
        private:
                float pay_per_month;
```

```
    public:
        void read_data() {
            Lecture::read_data();
            cout<<"Enter pay per month rate: ";
            cin>>pay_per_month;
        }
        void print_data() {
            Lecture::print_data();
            cout<<"Pay per month rate: "<<pay_per_month<<endl;
        }
};

int main() {
    PartTime p;
    FullTime f;

    cout<<"Enter part time teacher's information"<<endl;
    p.read_data();

    cout<<endl<<"Part time teacher's information: "<<endl;
    p.print_data();

    cout<<endl<<"Enter full time teacher's information"<<endl;
    f.read_data();

    cout<<endl<<"Full time teacher's information: "<<endl;
    f.print_data();

    return 0;
}
```

OUTPUT



```
Enter part time teacher's information
Enter ID and name of a lecture: 1 Suresh
Enter pay per hour rate: 10

Part time teacher's information:
ID: 1
Name: Suresh
Pay per hour rate: 10

Enter full time teacher's information
Enter ID and name of a lecture: 2 Ganesh
Enter pay per month rate: 1200

Full time teacher's information:
ID: 2
Name: Ganesh
Pay per month rate: 1200
```