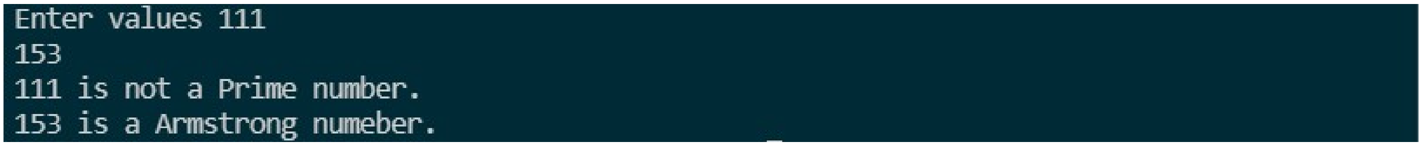
LAB - 8

Q1. Wap to create Base class Number. And make two derived class Prime & Armstrong. And display whether a number entered is prime or not and armstrong or not by declaring appropriate function.

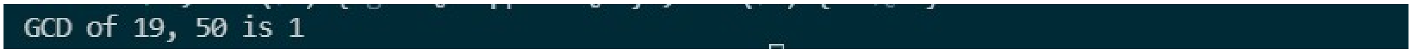
```
#include <iostream>
#include <math.h>
using namespace std;

class Number
{
protected:
    int a;

public:
    void get_data(int n)
    {
        a = n;
    }
};
class Prime : public Number
{
public:
    void isPrime()
    {
        int i, flag = 1, squareRoot;
        squareRoot = sqrt(a);
        for (i = 2; i <= squareRoot; ++i)
        {
            if (a % i == 0)
            {
                flag = 0;
                break;
            }
        }
        if (flag)
        {
            cout << a << " is a Prime number." << endl;
        }
        else
        {
            cout << a << " is not a Prime number." << endl;
        }
    }
};
class Armstrong : public Number
{
public:
    void isArmstrong()
    {
        int lastDigit, sum, originalNum, digits;
        sum = 0;
        originalNum = a;
        digits = (int)log10(a) + 1;
        while (a > 0)
        {
            lastDigit = a % 10;
            sum = sum + round(pow(lastDigit, digits));
            a = a / 10;
        }
        if (originalNum == sum)
        {
            cout << originalNum << " is a Armstrong numeber." << endl;
        }
        else
        {
            cout << originalNum << " is not a Armstrong numeber." << endl;
        }
    }
};

int main()
{
    int x,y;
    Prime p;
    cout<<"Enter values ";
    cin>>x>>y;
    p.get_data(x);
    p.isPrime();

    Armstrong a;
    a.get_data(y);
    a.isArmstrong();
    return 0;
}
```



```
Enter values 111
153
111 is not a Prime number.
153 is a Armstrong numeber.
```

Q2. Wap to create 2 base class A &B.having respective data member x, y.And create derived class GCD from A &B and display GCD of x and y using appropriate function.

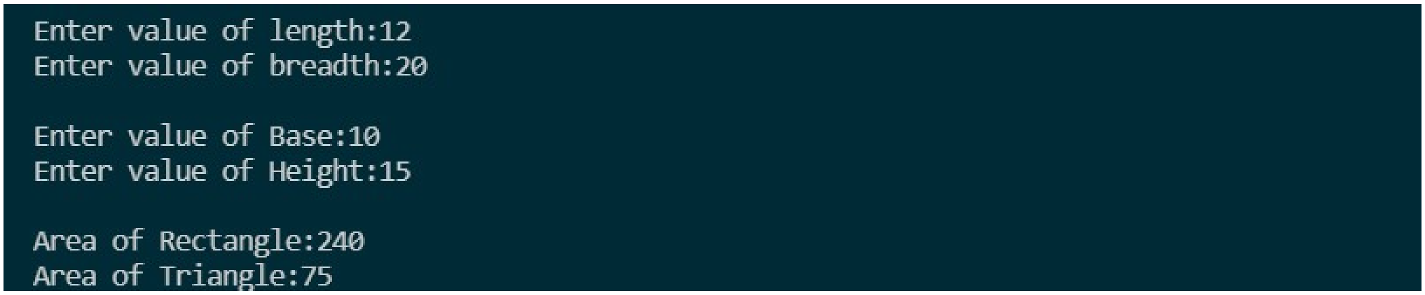
```
#include <iostream>
#include <math.h>
using namespace std;

class A
{
protected:
    int a;

public:
    void get_a(int n)
    {
        a = n;
    }
};
class B
{
protected:
    int b;

public:
    void get_b(int n)
    {
        b = n;
    }
};
class GCD : public A, public B
{
    int gcd;

public:
    void cal()
    {
        if (a > b)
        {
            int temp = a;
            a = b;
            b = temp;
        }
        for (int i = 1; i < b; i++)
        {
            if (a % i == 0 && b % i == 0)
                gcd = i;
        }
    }
    void display()
    {
        cout << "GCD of " << a << ", " << b << " is " << gcd << endl;
    }
};

int main()
{
    GCD var;
    var.get_a(50);
    var.get_b(19);
    var.cal();
    var.display();
}
```



```
GCD of 19, 50 is 1
```

Q3. Wap to create two base class named Rectange &Triange class.And common function getdata(), area().And another class Area derived from Rectange and Triangle with No member function.Display the area of Rectangle and Triangle.

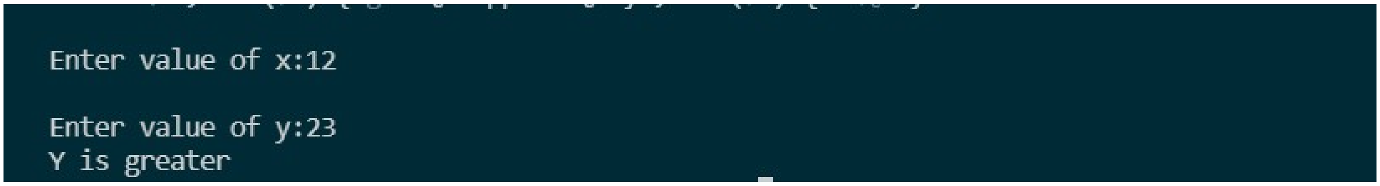
```
#include<iostream>

using namespace std;

class Rectangle
{
        protected:
                int a,b;
        public:
                 void getdata()
                 {
                           cout<<"\n Enter value of length:";
                           cin>>a;
               cout<<" Enter value of breadth:";
                           cin>>b;
                 }
                 void area()
                   {
                        int ar;
                        ar=(a*b);
                     cout<<"\n Area of Rectangle:"<<ar;
                   }
};
class Triangle
{
        protected:
                int ba,h;
        public:
                void get_data()
                {
                           cout<<"\n Enter value of Base:";
                           cin>>ba;
               cout<<" Enter value of Height:";
                           cin>>h;
                }
                void area()
                {
                        int ar;
                        ar=(ba*h)/2;
               cout<<"\n Area of Triangle:"<<ar;
                }
};
class AREA: public Rectangle , public Triangle
{
        public:

};

int main()
{
        AREA obj;
        obj.getdata();
        obj.get_data();
        obj.Rectangle::area();
        obj.Triangle::area();
        return 0;
    }
```



```
Enter value of length:12
Enter value of breadth:20

Enter value of Base:10
Enter value of Height:15

Area of Rectangle:240
Area of Triangle:75
```

Q4. Wap to create Base class A and data member is x.And B class derived from A having data member y.And C class derived from B will check greatest no between them using appropriate function.

```
    #include <iostream>

    using namespace std;

    class A
    {
    protected:
        int x;

    public:
        void getdata()
        {
            cout << "\n Enter value of x:";
            cin >> x;
        }
    };
    class B : public A
    {
    protected:
        int y;

    public:
        void get_data()
        {
            cout << "\n Enter value of y:";
            cin >> y;
        }
    };
    class C : public B
    {
    public:
        void calculate()
        {
            if (x > y)
                cout << " X is greater";
            else
            cout << " Y is greater";
        }
    };

    int main()
    {
        C obj;
        obj.getdata();
        obj.get_data();
        obj.calculate();
        return 0;
    }
```

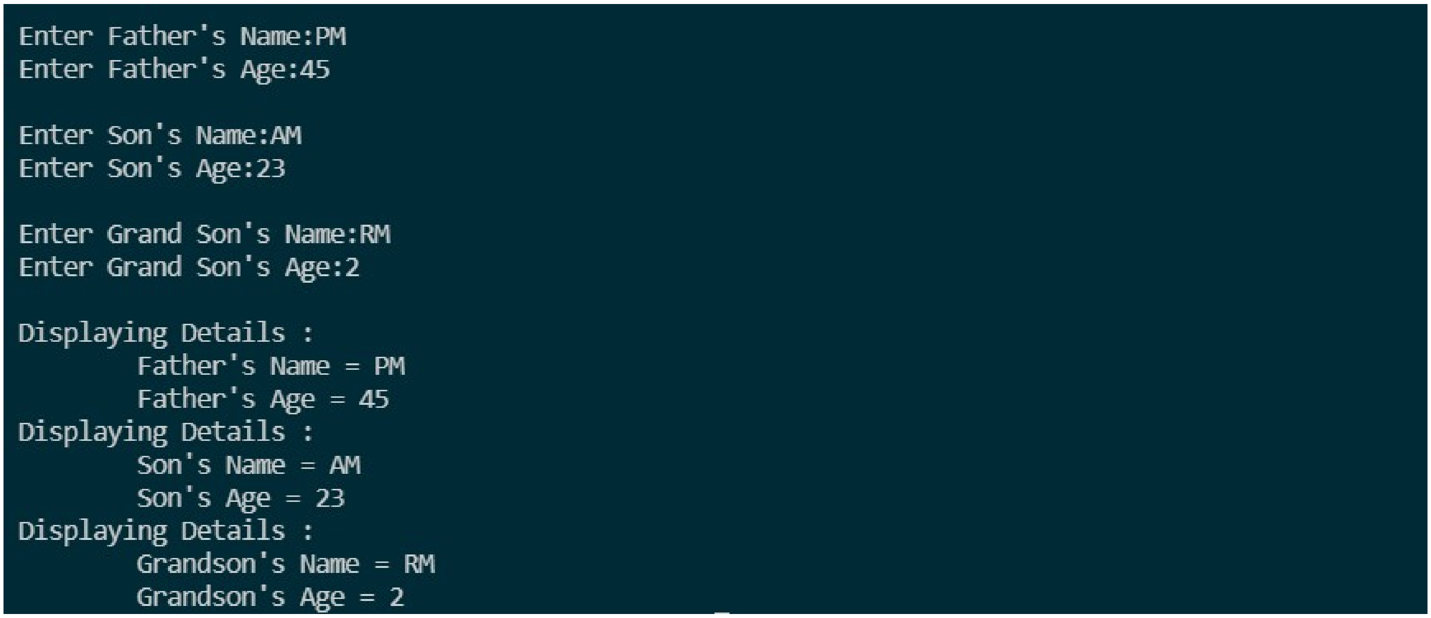


```
Enter value of x:12

Enter value of y:23
Y is greater
```

Q5. Wap to create father class having data member name and age.Another class son derived from father also having same data member.Grandson derived from son also having same data member.And common getdata putdata function.Display details of father, son and grandson.

```
    #include<iostream>
    #include<string>
    #include<stdio.h>
    using namespace std;
    class calculate;
    class Father
    {
            protected:
                    string name;
                    int age;
            public:
                     void getdata()
                     {
                              cout<<"\n Enter Father's Name:";
                              getline(cin,name);
                 cout<<" Enter Father's Age:";
                              cin>>age;
                     }
                     void putdata()
                     {
                              cout<<"\n Displaying Details :\n";
                          cout<<"\t Father's Name = "<<name;
                          cout<<"\n\t Father's Age = "<<age;
                     }
    };
    class Son: public Father
    {
            int age;
            string name;
            public:
                    void getdata()
```

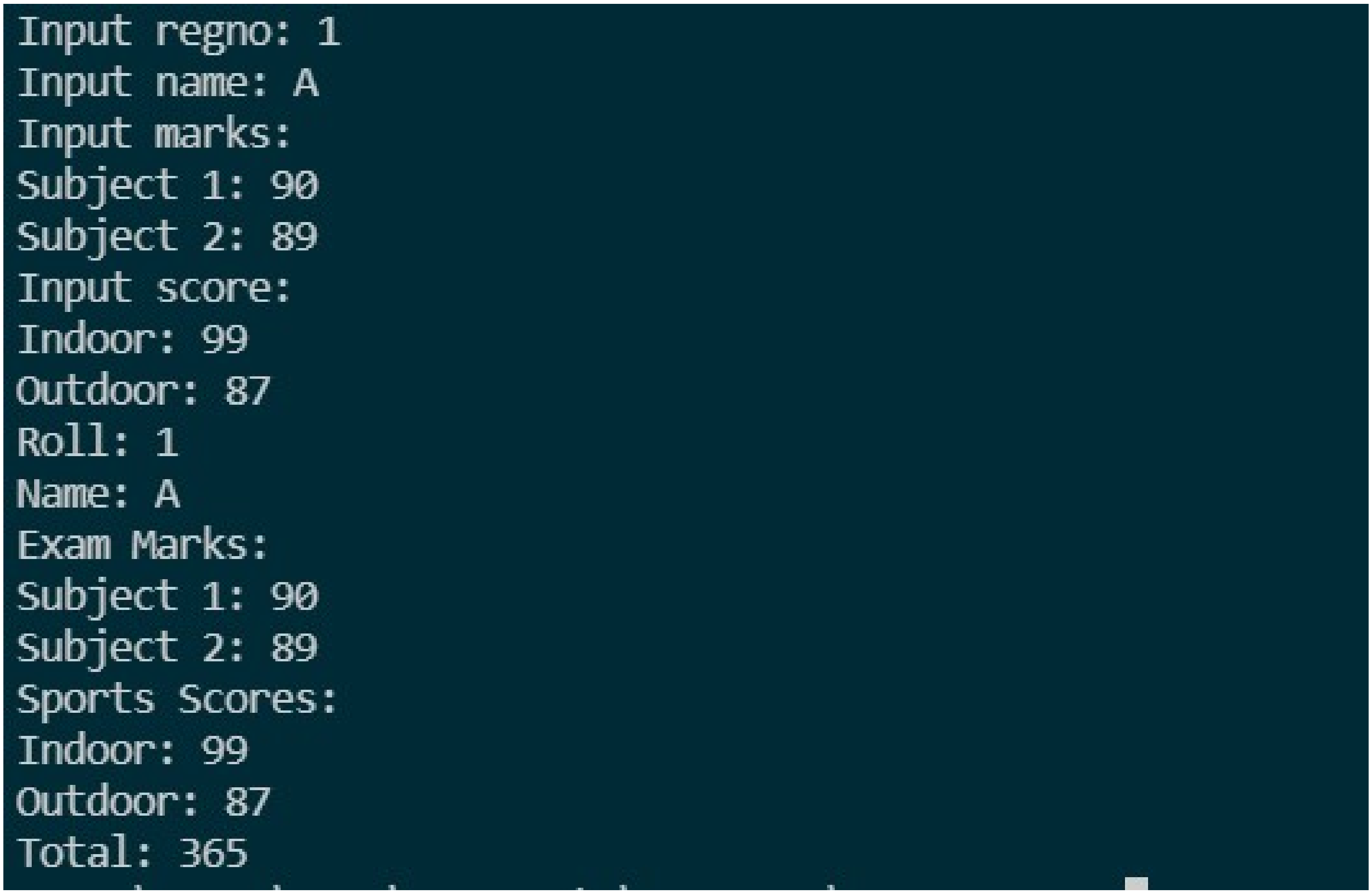
```
        {
            cout<<"\n Enter Son's Name:";
            fflush(stdin);
            getline(cin,name);
       cout<<" Enter Son's Age:";
             cin>>age;
        }
         void putdata()
         {
             cout<<"\n Displaying Details :\n";
            cout<<"\t Son's Name = "<<name;
            cout<<"\n\t Son's Age = "<<age;
         }

};
class Grand_Son: public Son
{
    int age;
    string name;
    public:
        void getdata()
        {
            cout<<"\n Enter Grand Son's Name:";
            fflush(stdin);
            getline(cin,name);
       cout<<" Enter Grand Son's Age:";
             cin>>age;
        }
         void putdata()
         {
             cout<<"\n Displaying Details :\n";
            cout<<"\t Grandson's Name = "<<name;
            cout<<"\n\t Grandson's Age = "<<age;
         }
};
int main()
{
    Grand_Son obj;
    obj.Father::getdata();
    obj.Son::getdata();
    obj.Grand_Son::getdata();
    obj.Father::putdata();
    obj.Son::putdata();
    obj.Grand_Son::putdata();
    return 0;
}
```



```
Enter Father's Name:PM
Enter Father's Age:45

Enter Son's Name:AM
Enter Son's Age:23

Enter Grand Son's Name:RM
Enter Grand Son's Age:2

Displaying Details :
        Father's Name = PM
        Father's Age = 45
Displaying Details :
        Son's Name = AM
        Son's Age = 23
Displaying Details :
        Grandson's Name = RM
        Grandson's Age = 2
```

Q6. Wap to create Student class.Data member name, roll.And Academic class derived from student and data member a1, a2 mark.And sport class derived from Student and data member is s1, s2 mark.And Result class derived from Academic &Sport, which will display roll, name, a1, a2 mark and s1, s2 mark along with display total mark using appropriate mark.

```
#include <iostream>
#include <string>
using namespace std;
class Student{
    protected:
        string name;
        int roll;
    public:
        Student(){}
        void getDetails(){
            cout<<"Input regno: ";
            cin>>roll;
            cout<<"Input name: ";
            cin>>name;
        }
        void putDetails(){
            cout<<"Roll: "<<roll<<endl;
            cout<<"Name: "<<name<<endl;
        }
};
class Exam: public Student{
    protected:
        int a1, a2;
    public:
        Exam(): Student(){}
        void getmarks(){
            cout<<"Input marks: \n";
            cout<<"Subject 1: ";
            cin>>a1;
            cout<<"Subject 2: ";
            cin>>a2;
        }
        void putmarks(){
            cout<<"Subject 1: "<<a1<<endl;
            cout<<"Subject 2: "<<a2<<endl;

        }
};
class Sports: public Student{
    protected:
        int indoor, outdoor, s1, s2;
    public:
        Sports(): Student(){}
        void getscore(){
            cout<<"Input score: \n";
            cout<<"Indoor: ";
            cin>>s1;
            cout<<"Outdoor: ";
            cin>>s2;
        }
        void putscore(){
            cout<<"Indoor: "<<s1<<endl;
            cout<<"Outdoor: "<<s2<<endl;
        }
};
class Result: public Sports, public Exam{
    int total;
    public:
        Result(): Sports(), Exam(){
            Exam::getDetails();
            getmarks();
            getscore();
            total = s1 + s2 + a1 + a2 ;
        }
        void display(){
            Exam::putDetails();
            cout<<"Exam Marks: \n"; Exam::putmarks();
            cout<<"Sports Scores: \n"; Sports::putscore();
            cout<<"Total: "<<total<<endl;
        }
};
int main(){
    Result result;
    result.display();
    return 0;
}
```



```
Input regno: 1
Input name: A
Input marks:
Subject 1: 90
Subject 2: 89
Input score:
Indoor: 99
Outdoor: 87
Roll: 1
Name: A
Exam Marks:
Subject 1: 90
Subject 2: 89
Sports Scores:
Indoor: 99
Outdoor: 87
Total: 365
```