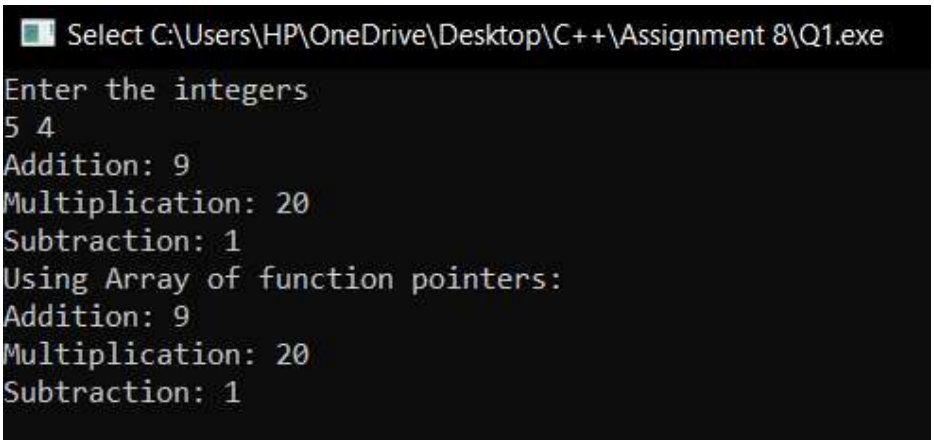
TUTORIAL 8

ARSHITA 21115024 P2

Q1

```
#include<iostream>
using namespace std;
int add(int x,int y)
{
        int sum=x+y;
        return sum;
}
int multiply(int x,int y)
{
        int prod=x*y;
        return prod;
}
int subtract(int x,int y)
{
        int diff=x-y;
        return diff;
}
int main()
{
        int a,b;
        cout<<"Enter the integers"<<endl;
        cin>>a>>b;
        int(*ptr)(int,int);
        ptr=&add;
        cout<<"Addition: "<<ptr(a,b)<<endl;
        ptr=&multiply;
        cout<<"Multiplication: "<<ptr(a,b)<<endl;
        ptr=&subtract;
```

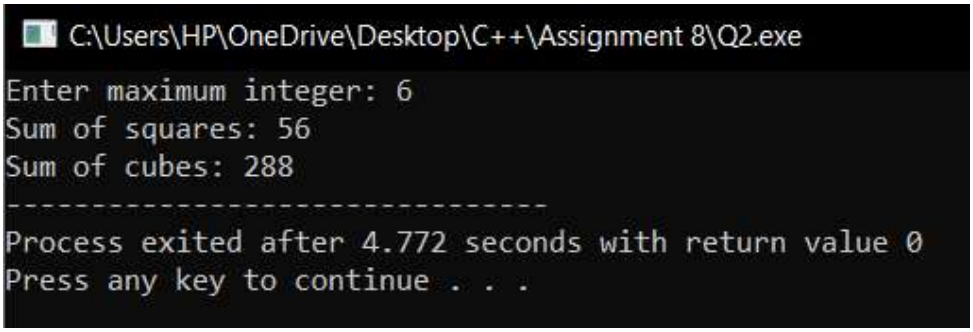
```
        cout<<"Subtraction: "<<ptr(a,b)<<endl;
        //using array of function pointers
        cout<<"Using Array of function pointers: "<<endl;
        int(*ptr1[3])(int,int)={add,multiply,subtract};
        cout<<"Addition: "<<ptr1[0](a,b)<<endl;
        cout<<"Multiplication: "<<ptr1[1](a,b)<<endl;
        cout<<"Subtraction: "<<ptr1[2](a,b)<<endl;
}
```



Q2

```
#include<iostream>
using namespace std;
int square(int a)
{
        return a*a;
}
int cube(int b)
{
        return b*b*b;
}
int sum(int(*)(int),int);
int main()
{
        int n;
```

```
        cout<<"Enter maximum integer: ";
        cin>>n;
        cout<<"Sum of squares: "<<sum(square,n)<<endl;
        cout<<"Sum of cubes: "<<sum(cube,n);
}
sum(int(*pf)(int),int b)
{
        int sum=0;
        for(int i=0;i<=b;i=i+2)
        {
                sum=sum+(*pf)(i);
        }
        return sum;
}
```



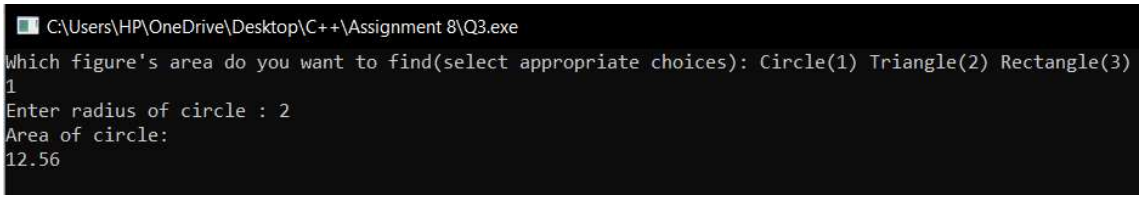
Q3

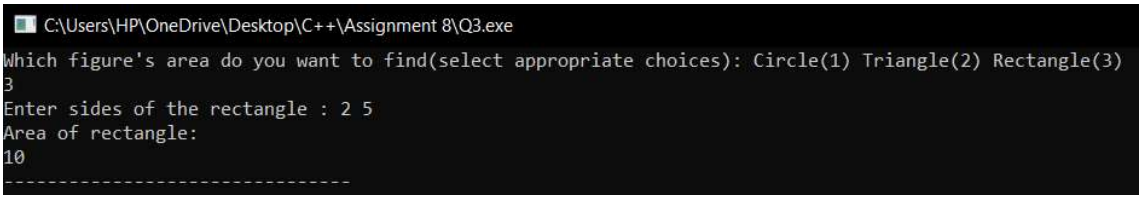
```
#include<iostream>
#include<cmath>
using namespace std;
class area
{
        double radius;
        double side1,side2;
        double a,b,c;
        public:
//for circle
```

```
area(double r)
{
        radius=r;
}
//for a rectangle
area(double s1,double s2)
{
side1=s1;
side2=s2;
}
//for triangle
area(double st1,double st2,double st3)
{
a=st1;
b=st2;
c=st3;
}
//circle area function
void circle()
{
        double circleArea=(3.14)*radius*radius;
        cout<<"Area of circle: "<<endl;
        cout<<circleArea<<endl;
}
//rectangle area function
void rectangle()
{
        double rectarea=side1*side2;
        cout<<"Area of rectangle: "<<endl;
        cout<<rectarea;
}
```

```
//triangle area function
void triangle()
{
        double s=(a+b+c)/2;
        double v=s*(s-a)*(s-b)*(s-c);
        double triarea=sqrt(v);
        cout<<"Area of triangle: "<<endl;
        cout<<triarea<<endl;
}
};
int main()
{
cout<<"Which figure's area do you want to find(select appropriate choices): Circle(1) Triangle(2) Rectangle(3)"<<endl;
int ch;
cin>>ch;
if(ch==1)
{
        cout<<"Enter radius of circle : ";
        int r;
        cin>>r;
   area s1(r);
   s1.circle();
}
if(ch==3)
{
        cout<<"Enter sides of the rectangle : ";
        int a,b;
        cin>>a>>b;
   area s2(a,b);
   s2.rectangle();
```

```
}
if(ch==2)
{
        cout<<"Enter sides of the triangle : ";
        int x,y,z;
        cin>>x>>y>>z;
   area s3(x,y,z);
   s3.triangle();
}
}
```



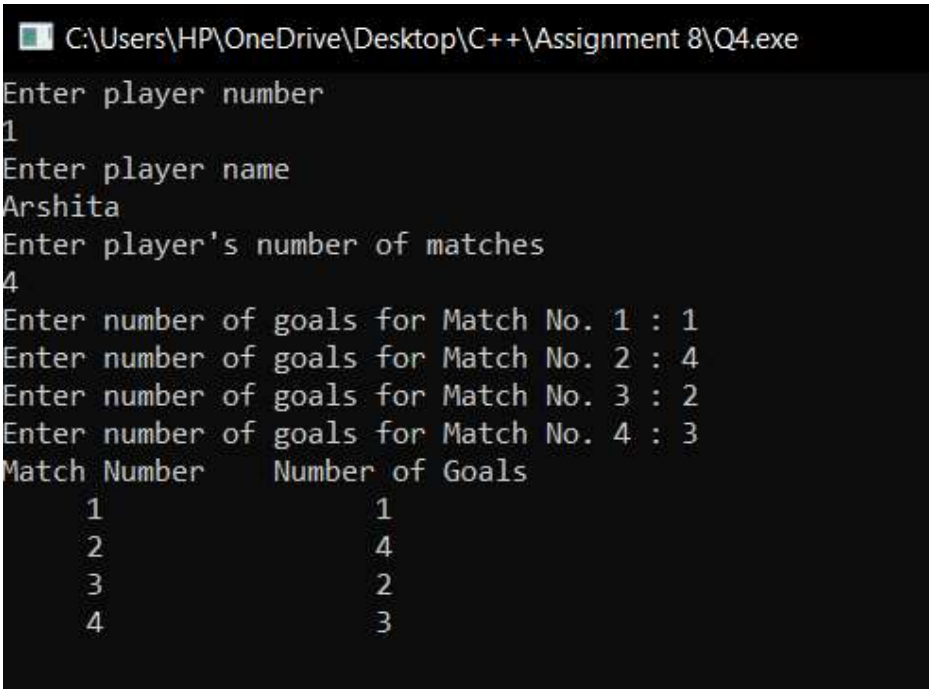


Q4

```
#include<iostream>
#include<string>
using namespace std;
class player
{
public:
int playnum;
string playname;
```

```
int nummatch;
int *arrayplayer;
player(int playnum,string playname,int nummatch)
  {
        this->nummatch=nummatch;
        this->playname=playname;
        this->playnum=playnum;
        arrayplayer= new int[this->nummatch];
        for(int i=0;i<nummatch;i++)
        {
              cout<<"Enter number of goals for Match No. "<<i+1<<" : ";
              cin>>arrayplayer[i];
        }
        }
void display()
{
cout<<"Match Number"<<"    "<<"Number of Goals"<<endl;
for(int i=0;i<nummatch;i++)
{
cout<<"    "<<i+1<<"       "<<"      "<<arrayplayer[i]<<" "<<endl;
}
}
~player()
{
delete[]arrayplayer;
arrayplayer=NULL;
}
};
int main()
{
int playnum;
```

```
string playname;
int nummatch;
cout<<"Enter player number"<<endl;
cin>>playnum;
cout<<"Enter player name"<<endl;
cin>>playname;
cout<<"Enter player's number of matches"<<endl;
cin>>nummatch;
player p1(playnum,playname,nummatch);
p1.display();
}
```

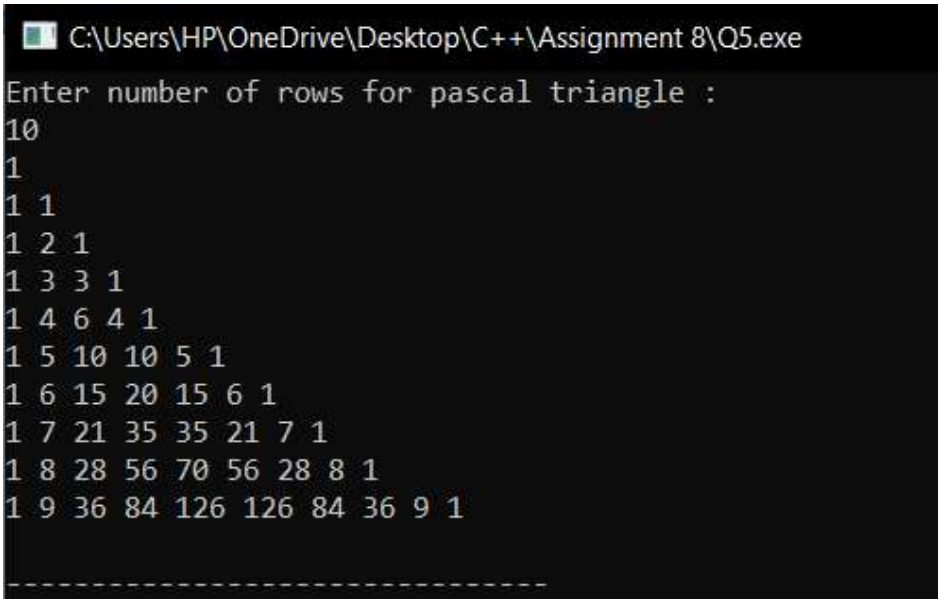


Q5

```
#include<iostream>
using namespace std;
class pascal
{
private:
int n;
```

```
public:
        void setvalue(int n)
        {
                this->n=n;
        }
        int bincoeff(int n,int k)
        {
                int f=1;
                if(k>n-k)
                {
                        k=n-k;
                }
                for(int i=0;i<k;i++)
                {
                f=f*(n-i);
                f=f/(i+1);
                }
                return f;
        }
friend void pascalseries(pascal);
};
void pascalseries(pascal h)
{
        h.n=h.n-1;
        cout<<1<<endl;
for(int i=1;i<=h.n;i++)
{
        cout<<1<<" ";
        for(int j=1;j<=i;j++)
        {
                cout<<h.bincoeff(i,j)<<" ";
```

```
        }
        cout<<endl;
}
}
int main()
{
        int n;
        cout<<"Enter number of rows for pascal triangle : "<<endl;
        cin>>n;
        pascal b1;
        b1.setvalue(n);
        pascalseries(b1);
}
```



Q6

```
#include<iostream>
using namespace std;
//fibonacci series
//friend functions
class fibonacci
{
```

```
private:
int n;
int c1=0;
int c2=1;
public:
        void setvalue(int n)
        {
                this->n=n;
        }
int operator+()
{
return (c1+c2);
}
friend void fibbfunc(fibonacci);
};
void fibbfunc(fibonacci h)
{
        cout<<h.c1<<" "<<h.c2<<" ";
   for(int i=3;i<=h.n;i++)
   {
        int c3=h.c1+h.c2;
        cout<<c3<<" ";
        int f=h.c2;
        h.c2=c3;
        h.c1=f;
        }
}

int main()
{
        int n;
```

```
        cout<<"Enter number of elements in series : "<<endl;
        cin>>n;
        fibonacci b1;
        b1.setvalue(n);
        fibbfunc(b1);
}
```

```
C:\Users\HP\OneDrive\Desktop\C++\Assignment 8\Q6.exe
Enter number of elements in series :
15
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377
--------------------------------
Process exited after 3.75 seconds with return value 0
Press any key to continue . . .
```

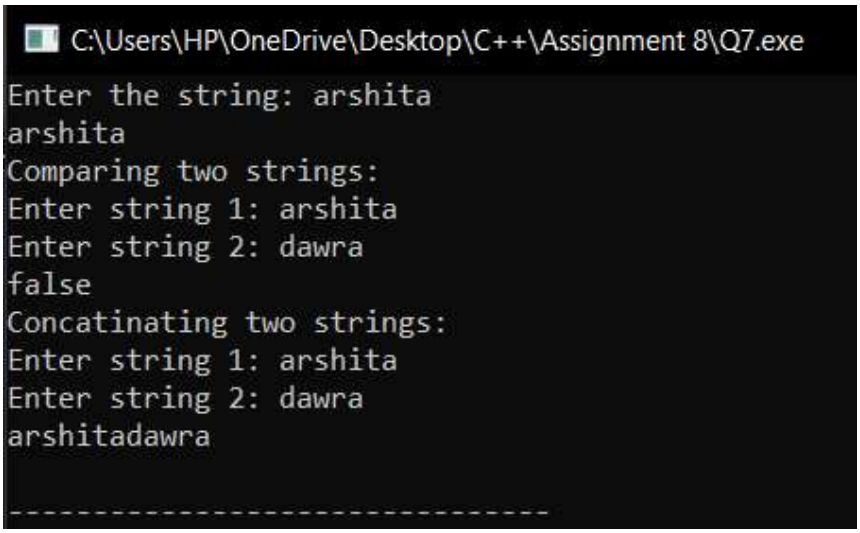
Q7

```
#include<iostream>
#include<string>
using namespace std;
class strings
{
   string sm;
   string s1;
   string s2;
   string sc1;
   string sc2;
   public:
   void setstring1(string input)
   {
      sm=input;
   }
   void display()
   {
```

```
        cout<<sm<<endl;
    }
     void setstring2(string ss1,string ss2)
    {
        s1=ss1;
        s2=ss2;
    }
    void check()
    {
        if(s1==s2)
        {
           cout<<"true"<<endl;
        }
        else
        {
          cout<<"false"<<endl;
        }
    }
   void setstring3(string ssc1,string ssc2)
    {
        sc1=ssc1;
        sc2=ssc2;
    }
    void concat()
    {
        string c=sc1+sc2;
        cout<<c<<endl;
    }
};

int main()
```

```
{
   string s;
   cout<<"Enter the string: ";
   cin>>s;
   strings s1;
   s1.setstring1(s);
   s1.display();
   cout<<"Comparing two strings: "<<endl;
   cout<<"Enter string 1: ";
   string string1;
   cin>>string1;
   cout<<"Enter string 2: ";
   string string2;
   cin>>string2;
   strings s2;
   s2.setstring2(string1,string2);
   s2.check();
   cout<<"Concatinating two strings: "<<endl;
   cout<<"Enter string 1: ";
   string stringg1;
   cin>>stringg1;
   cout<<"Enter string 2: ";
   string stringg2;
   cin>>stringg2;
   strings s3;
   s3.setstring3(stringg1,stringg2);
   s3.concat();
   return 0;
}
```



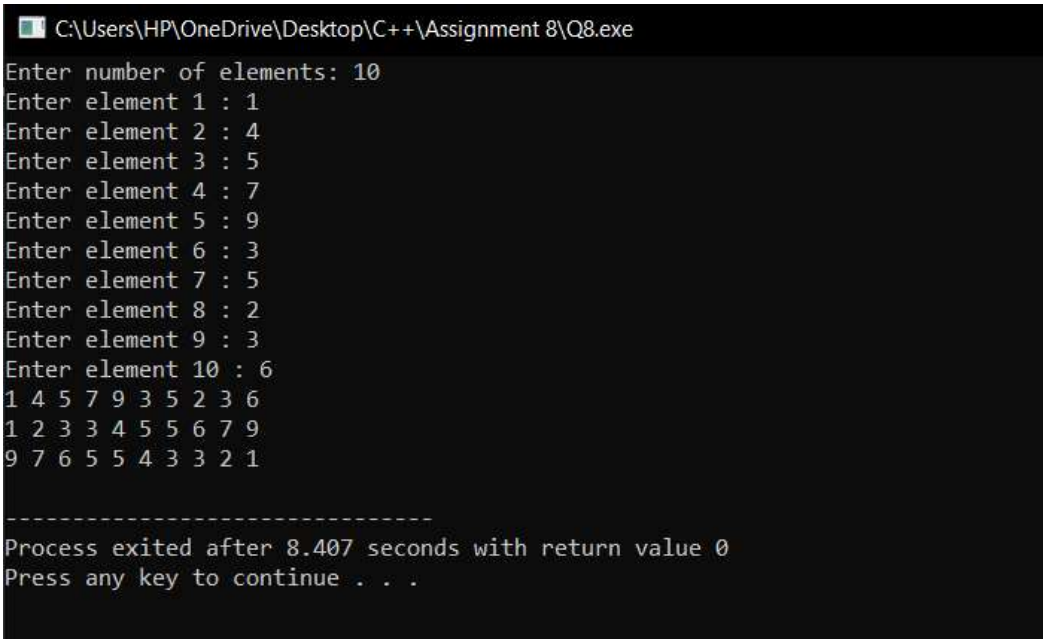
Q8

```
#include<iostream>
using namespace std;
class node
{
public:
	int data;
	node* next;
	//constructor
	node(int val)
	{
		data=val;
		next=NULL;
	}
};
void insertAtTail(node* &head,int val)
{
	//forming a new node
	node* n=new node(val);
	if(head==NULL)
	{
```

```
                head=n;
                return;
        }
        node* temp=head;
        //traversing to end
        while(temp->next!=NULL)
        {
                temp=temp->next;
        }
        temp->next=n;
}
void display(node*head)
{
        //displaying the list
        node*temp=head;
        while(temp!=NULL)
        {
                cout<<temp->data<<" ";
                temp=temp->next;
        }
        cout<<endl;
}
void ascendingsorting(node* head)
{
        node *i;
        node *j;
        for(i=head;i!=NULL;i=i->next)
        {
                for(j=i->next;j!=NULL;j=j->next)
                {
                 if((i->data)>(j->data))
```

```
                    {
                      int c=i->data;
                      i->data=j->data;
                      j->data=c;
                    }
                   }
          }
}
void descendingsorting(node* head)
{
          node *i;
          node *j;
          for(i=head;i!=NULL;i=i->next)
          {
                    for(j=i->next;j!=NULL;j=j->next)
                    {
                     if((i->data)<(j->data))
                     {
                      int c=i->data;
                      i->data=j->data;
                      j->data=c;
                     }
                    }
          }
}
int main()
{
//creating empty list
node* head=NULL;
cout<<"Enter number of elements: ";
int g;
```

```
cin>>g;
for(int i=1;i<=g;i++)
{
        cout<<"Enter element "<<i<<" : ";
        int n;
        cin>>n;
   insertAtTail(head,n);
}
//displaying original linked list
display(head);
//sorting in ascending order
ascendingsorting(head);
//displaying sorted list
display(head);
//sorting in descending order
descendingsorting(head);
//displaying sorted list
display(head);
}
```

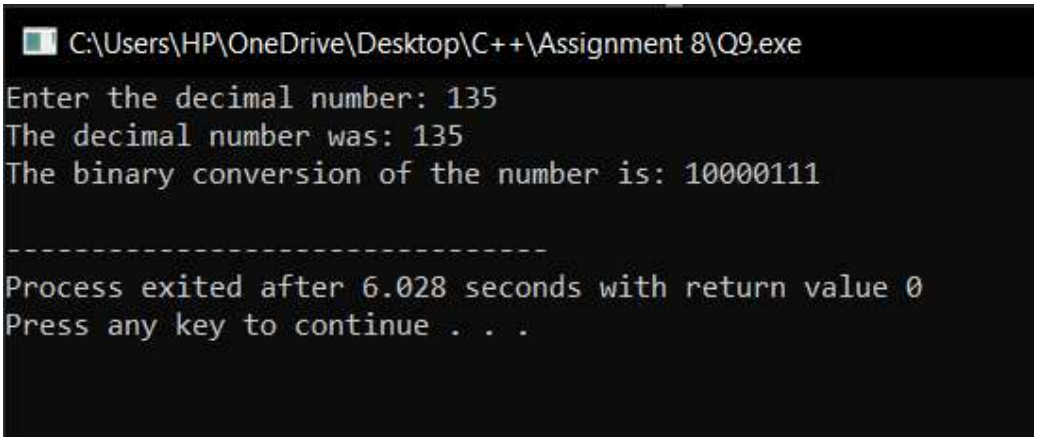


Q9

```
#include<iostream>
#include<string>
using namespace std;
//using multiple inheritance
class decimal
{
public:
int n;
void setValue(int n)
{
        this->n=n;
}
};
class binary
{
public:
int c;
string str;
void convert(int n)
{
        str="";
   if(n==0)
   {
      str="0";
   }
   else
        {
   while(n>0)
   {
     c=n%2;
```

```
        n=n/2;
        if(c==1)
        {
         str="1"+str;
        }
        else
        {
           str="0"+str;
        }
    }
    }
}
};
class derived: public decimal, public binary
{
public:
void print()
   {
            cout<<"The decimal number was: "<<n<<endl;
            cout<<"The binary conversion of the number is: "<<str<<endl;
            }
};
int main()
{
            int n;
            cout<<"Enter the decimal number: ";
    cin>>n;
    derived a1;
    a1.setValue(n);
    a1.convert(n);
    a1.print();
```

}

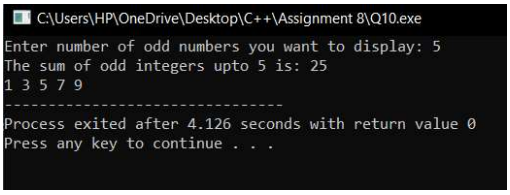


Q10

```
#include<iostream>
using namespace std;
class oddNum
{
public:
        int n;
void setValue()
  {
        cout<<"Enter number of odd numbers you want to display: ";
        cin>>n;
        }
};
class sum
{
public:
int sum;
void sumcalc(int n)
{
        sum=0;
        int f=1;
```

```
for(int i=1;f<=n;i=i+2)
   {
          sum=sum+i;
          f++;
          }
}
};
class display
{
public:
void displayNum(int n)
{
int f=1;
for(int i=1;f<=n;i=i+2)
   {
          cout<<i<<" ";
          f++;
          }
}
};
class derived:public sum, public display,public oddNum
{
public:
void sumdisplay(void)
{
cout<<"The sum of odd integers upto "<<n<<" is: "<<sum<<endl;
}
};
int main()
{
derived a1;
```

```
a1.setValue();
int v=a1.n;
a1.sumcalc(v);
a1.sumdisplay();
a1.displayNum(v);
}
```



Q11

```
#include<iostream>
#include<cmath>
using namespace std;
class fixDeposit
{
protected:
int accno;// account number
protected:
double amount;// principal amount
public:
fixDeposit(int a, double p)
{
accno = a;
amount = p;
}
public:
double interest()
{
cout << "The member function in fixDeposit";
return 0;
```

```
}
public:
void update(double d)
{
amount += d;
}
public:
void display()
{
cout<<"The member function in fixDeposit";
}
};
class SIdeposit: public fixDeposit
{
public:
double rateinterest;
int timeyears;
double interestSI;
SIdeposit(int a,double b,double c,int d):fixDeposit(a,b)
{
        rateinterest=c;
        timeyears=d;
}
double interest()
{
   interestSI=(amount*rateinterest*timeyears)/100;
        return interestSI;
}
void update()
{
   amount += interestSI;
```

```
}

void display()
{
   cout<<"Updated Amount: "<<amount<<endl;
}

};
class CIdeposit: public fixDeposit
{
public:
double rateinterest;
int timeyears;
double interestCI;
CIdeposit(int a,double b,double c,int d):fixDeposit(a,b)
{
        rateinterest=c;
        timeyears=d;
}
double interest()
{
        interestCI=amount*(pow(1+(rateinterest/100),timeyears))-amount;
        return interestCI;
}
void update()
{
   amount += interestCI;
}

void display()
{
```

```
   cout<<"Updated Amount: "<<amount<<endl;
}
};
int main()
{
        cout<<"Enter account number: ";
        int a;
        cin>>a;
        cout<<"Enter principle amount: ";
        int b;
        cin>>b;
        cout<<"Enter yearly rate of interest for SI: ";
        int c1;
        cin>>c1;
        cout<<"Enter time period of deposit in number of years: ";
        int d1;
        cin>>d1;
        cout<<"Enter yearly rate of interest for CI: ";
        int c2;
        cin>>c2;
        cout<<"Enter time period of deposit in number of years: ";
        int d2;
        cin>>d2;
        SIdeposit u1(a,b,c1,d1);
        SIdeposit * SIptr;
        SIptr=&u1;
        cout<<"The SI is: "<<SIptr->interest()<<endl;
        SIptr->update();
        SIptr->display();
        CIdeposit u2(a,b,c2,d2);
        CIdeposit * CIptr;
```

```
        CIptr=&u2;
        cout<<"The CI is: "<<CIptr->interest()<<endl;
        CIptr->update();
        CIptr->display();
}
```

```
C:\Users\HP\OneDrive\Desktop\C++\Assignment 8\Q11.exe
Enter account number: 1
Enter principle amount: 100
Enter yearly rate of interest for SI: 4
Enter time period of deposit in number of years: 2
Enter yearly rate of interest for CI: 4
Enter time period of deposit in number of years: 2
The SI is: 8
Updated Amount: 108
The CI is: 8.16
Updated Amount: 108.16
```

Q12

```
#include<iostream>
#include<cmath>
using namespace std;
class fixDeposit
{
public:
static int accno;// account number
protected:
double amount;// principal amount
public:
fixDeposit(double p)
{
amount = p;
}
public:
double interest()
{
```

```
cout << "The member function in fixDeposit";
return 0;
}
public:
void update(double d)
{
amount += d;
}
public:
void display()
{
cout<<"The member function in fixDeposit";
}
};
int fixDeposit::accno;
class SIdeposit: public fixDeposit
{
public:
double rateinterest;
int timeyears;
double interestSI;
SIdeposit(double b,double c,int d):fixDeposit(b)
{
        rateinterest=c;
        timeyears=d;
}
double interest()
{
   interestSI=(amount*rateinterest*timeyears)/100;
        return interestSI;
}
```

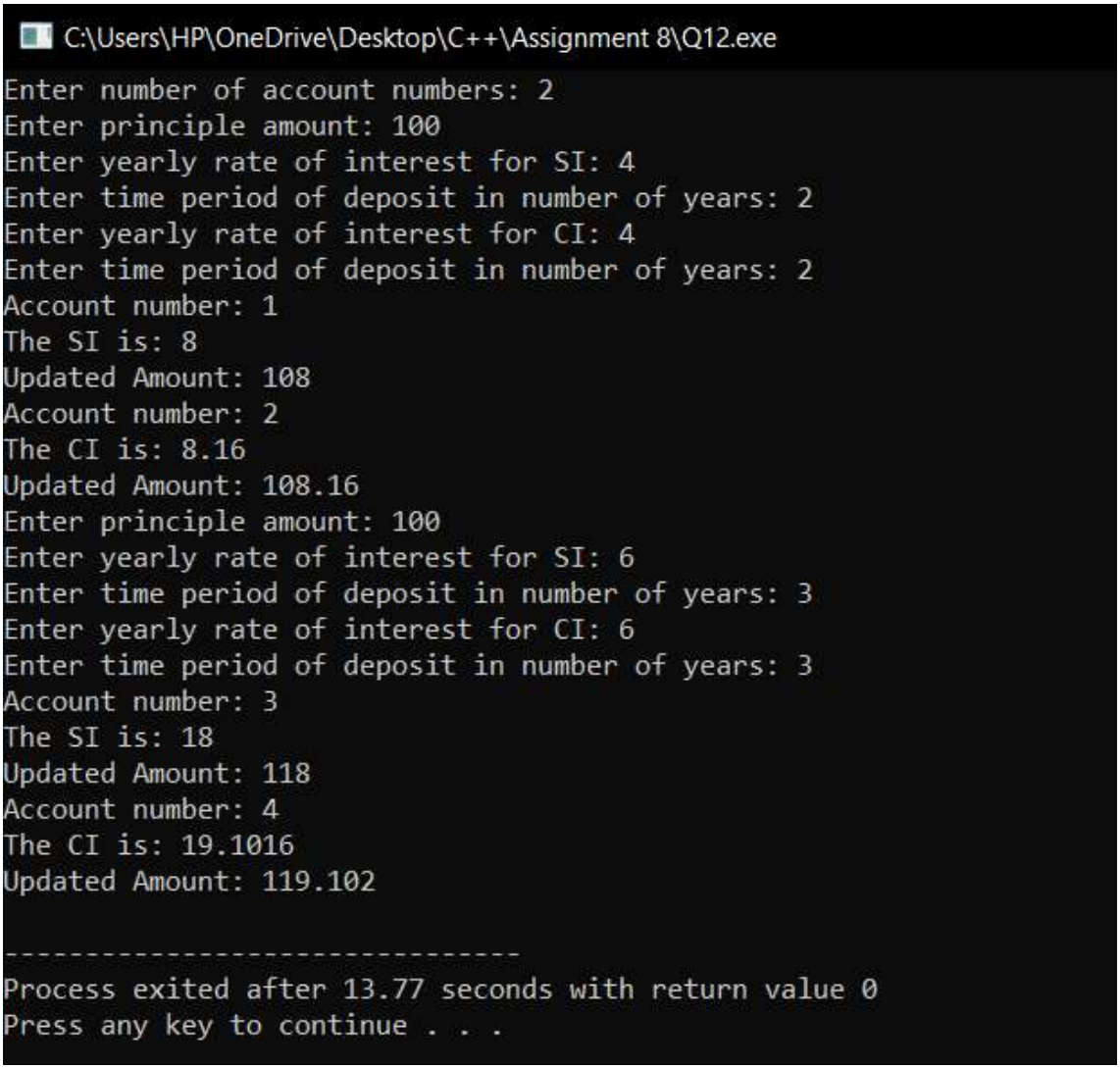
```
void update()
{
   amount += interestSI;
}

void display()
{
   cout<<"Updated Amount: "<<amount<<endl;
}

};
class CIdeposit: public fixDeposit
{
public:
double rateinterest;
int timeyears;
double interestCI;
CIdeposit(double b,double c,int d):fixDeposit(b)
{
        rateinterest=c;
        timeyears=d;
}
double interest()
{
        interestCI=amount*(pow(1+(rateinterest/100),timeyears))-amount;
        return interestCI;
}
void update()
{
   amount += interestCI;
}
```

```
void display()
{
   cout<<"Updated Amount: "<<amount<<endl;
}
};
int main()
{
        cout<<"Enter number of account numbers: ";
        int f;
        cin>>f;
        for(int i=1;i<=f;i++)
        {
        cout<<"Enter principle amount: ";
        int b;
        cin>>b;
        cout<<"Enter yearly rate of interest for SI: ";
        int c1;
        cin>>c1;
        cout<<"Enter time period of deposit in number of years: ";
        int d1;
        cin>>d1;
        cout<<"Enter yearly rate of interest for CI: ";
        int c2;
        cin>>c2;
        cout<<"Enter time period of deposit in number of years: ";
        int d2;
        cin>>d2;
        SIdeposit u1(b,c1,d1);
        cout<<"Account number: "<<++(u1.accno)<<endl;
        SIdeposit * SIptr;
```

```
        SIptr=&u1;
        cout<<"The SI is: "<<SIptr->interest()<<endl;
        SIptr->update();
        SIptr->display();
        CIdeposit u2(b,c2,d2);
        cout<<"Account number: "<<++(u2.accno)<<endl;
        CIdeposit * CIptr;
        CIptr=&u2;
        cout<<"The CI is: "<<CIptr->interest()<<endl;
        CIptr->update();
        CIptr->display();
}
}
```



Q13

```
#include<iostream>

#include<cmath>

using namespace std;

class fixDeposit

{

protected:

int accno;// account number

protected:

double amount;// principal amount

public:
```

```
fixDeposit(int a, double p)
{
accno = a;
amount = p;
}
public:
virtual double interest()
{
cout << "The member function in fixDeposit";
return 0;
}
public:
void update(double d)
{
amount += d;
}
public:
virtual void display()
{
cout<<"The member function in fixDeposit";
}
};
class SIdeposit: public fixDeposit
{
public:
double rateinterest;
int timeyears;
double interestSI;
SIdeposit(int a,double b,double c,int d):fixDeposit(a,b)
{
        rateinterest=c;
```

```
        timeyears=d;
}
double interest()
{
   interestSI=(amount*rateinterest*timeyears)/100;
        return interestSI;
}
void display()
{
        update(interestSI);
   cout<<"Updated Amount: "<<amount<<endl;
}
};
class CIdeposit: public fixDeposit
{
public:
double rateinterest;
int timeyears;
double interestCI;
CIdeposit(int a,double b,double c,int d):fixDeposit(a,b)
{
        rateinterest=c;
        timeyears=d;
}
double interest()
{
        interestCI=amount*(pow(1+(rateinterest/100),timeyears))-amount;
        return interestCI;
}
void display()
{
```

```
        update(interestCI);
   cout<<"Updated Amount: "<<amount<<endl;
}
};
int main()
{
        cout<<"Enter account number: ";
        int a;
        cin>>a;
        cout<<"Enter principle amount: ";
        int b;
        cin>>b;
        cout<<"Enter yearly rate of interest for SI: ";
        int c1;
        cin>>c1;
        cout<<"Enter time period of deposit in number of years: ";
        int d1;
        cin>>d1;
        cout<<"Enter yearly rate of interest for CI: ";
        int c2;
        cin>>c2;
        cout<<"Enter time period of deposit in number of years: ";
        int d2;
        cin>>d2;
        SIdeposit u1(a,b,c1,d1);
        fixDeposit * SIptr;
        SIptr=&u1;
        cout<<"The SI is: "<<SIptr->interest()<<endl;
        SIptr->display();
        CIdeposit u2(a,b,c2,d2);
        fixDeposit * CIptr;
```

```
        CIptr=&u2;
        cout<<"The CI is: "<<CIptr->interest()<<endl;
        CIptr->display();
}
```

```
C:\Users\HP\OneDrive\Desktop\C++\Assignment 8\Q13.exe
Enter account number: 1
Enter principle amount: 100
Enter yearly rate of interest for SI: 4
Enter time period of deposit in number of years: 2
Enter yearly rate of interest for CI: 4
Enter time period of deposit in number of years: 2
The SI is: 8
Updated Amount: 108
The CI is: 8.16
Updated Amount: 108.16

--------------------------------
Process exited after 9.271 seconds with return value 0
Press any key to continue . . .
```

Yes the results remain same.

In one case we use pointer to derived class.

In one case we use pointer of base class, by making functions of base class virtual.