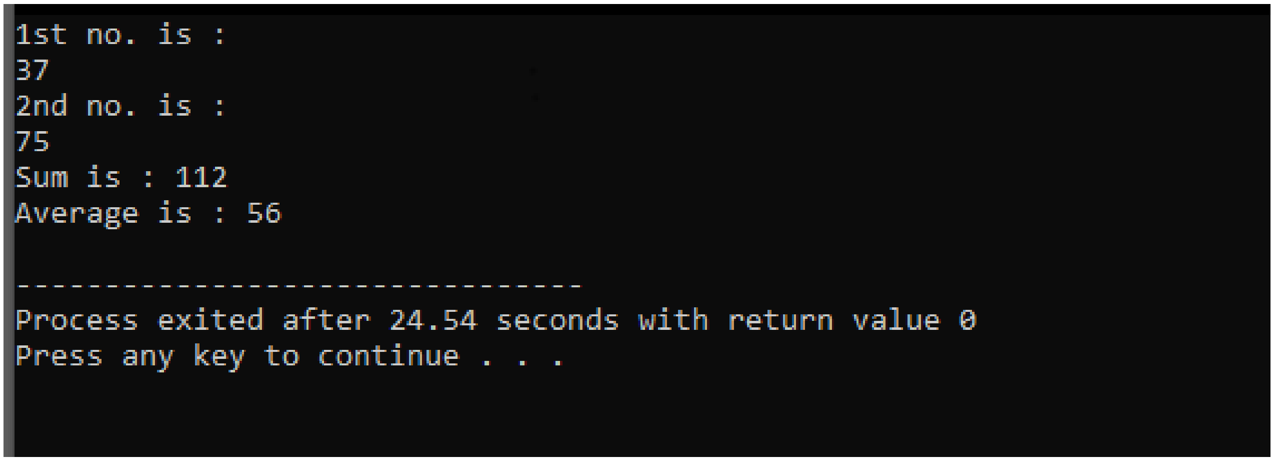
LAB TASK No. 1

Task 1: Write a program that read two integers from the keyboard and print their sum and average.

Code:

```
#include<conio.h>
#include<iostream>
#include<string.h>
#include<math.h>
using namespace std;
int main()
 {
        int num1,num2;
        cout<<"1st no. is : "<<endl;
        cin>>num1;
        cout<<"2nd no. is : "<<endl;
        cin>>num2;
        int sum;
        sum = num1 + num2;
        cout<<"Sum is : "<<sum<<endl;
        double avg;
                          avg = (num1 + num2)/2;
        cout<<"Average is : "<<avg<<endl;
        return 0;
}
```

Output:



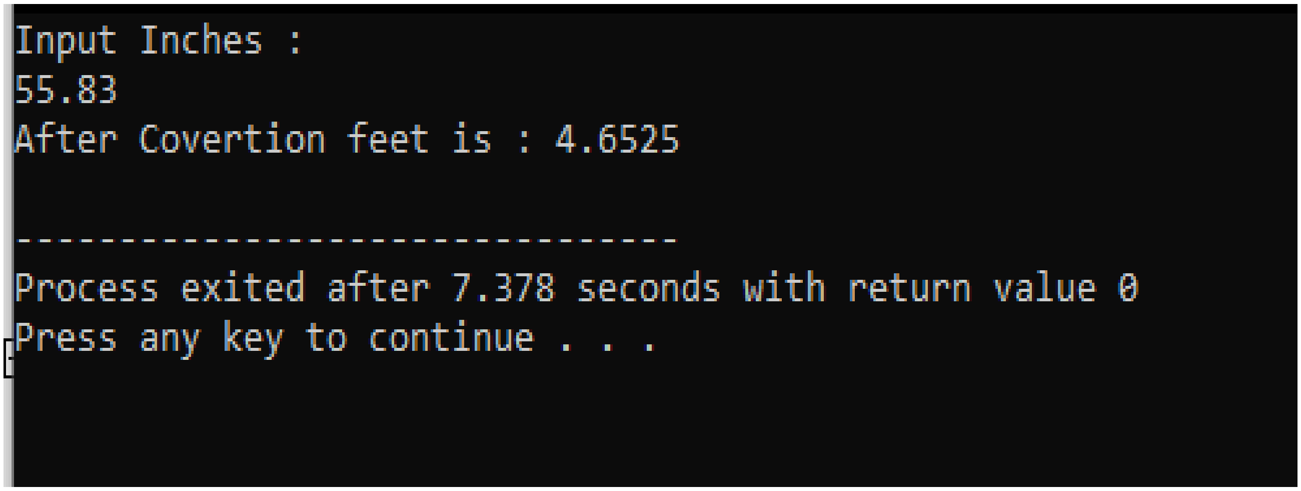
```
1st no. is :
37
2nd no. is :
75
Sum is : 112
Average is : 56

--------------------------------
Process exited after 24.54 seconds with return value 0
Press any key to continue . . .
```

Task 2: Write a program that prompts for a person's height in inches. Convert this height measurement into feet by using the conversion factor of foot2Inch= 12 inch. Now, the value obtained can easily, be translated into feet and inches which are then output by the program.

Code:

```
#include<conio.h>
#include<iostream>
#include<string.h>
#include<math.h>
using namespace std;
double feetinches(double i)
{
    double f = i/12;
    return f;
 }
int main()
 {
    double ch;
     cout<<"Input Inches :"<<endl;
     cin>> ch;
     cout<<"After Covertion feet is : "<<feetinches(ch)<<endl;
    return 0;
}
```

Output:



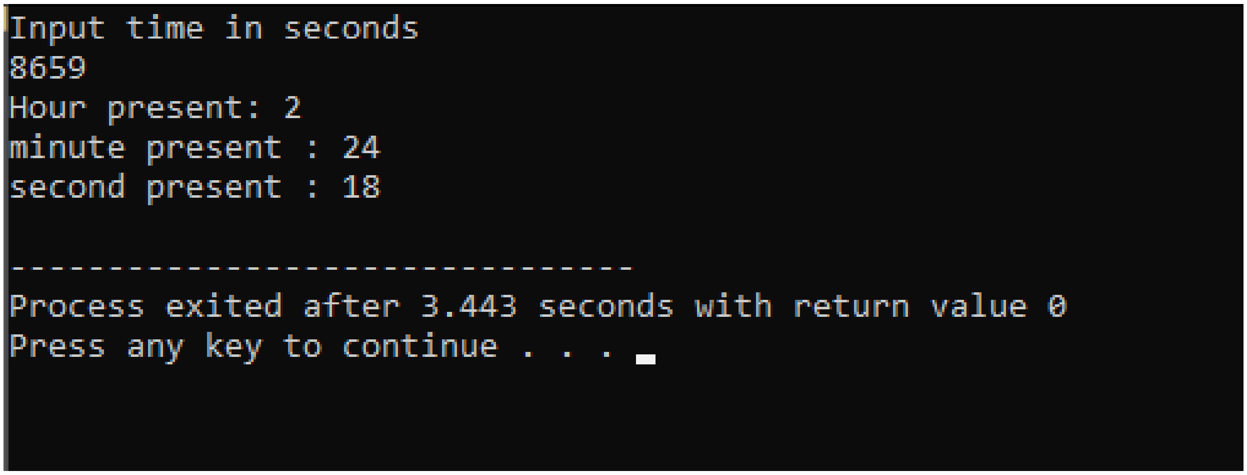
```
Input Inches :
55.83
After Covertion feet is : 4.6525


--------------------------------
Process exited after 7.378 seconds with return value 0
Press any key to continue . . .
```

Task 3: Write a program that prompts for time in seconds and output that time in hours, minutes, and seconds. Here student will learn the usage of divide and modulus arithmetic operators in integers.

Code:

```
#include<conio.h>
#include<iostream>
#include<string.h>
#include<math.h>
using namespace std;
void hms(double s)
{
    double i1,i2;
    double hr = s/3600;
   double fr1=modf(hr,&i1);
    double min = fr1 * 60 ;
   double fr2=modf(min,&i2);
    int sec=fr2 * 60;
    cout<<"Hour present: "<<i1<<endl;
    cout<<"minute present : "<<i2<<endl;
    cout<<"second present : "<<sec<<endl;
}
int main()
{
        double sec;

        cout<<"Input time in seconds "<<endl;
        cin>>sec;
        hms(sec);
             return 0 ;
}
```

Output:



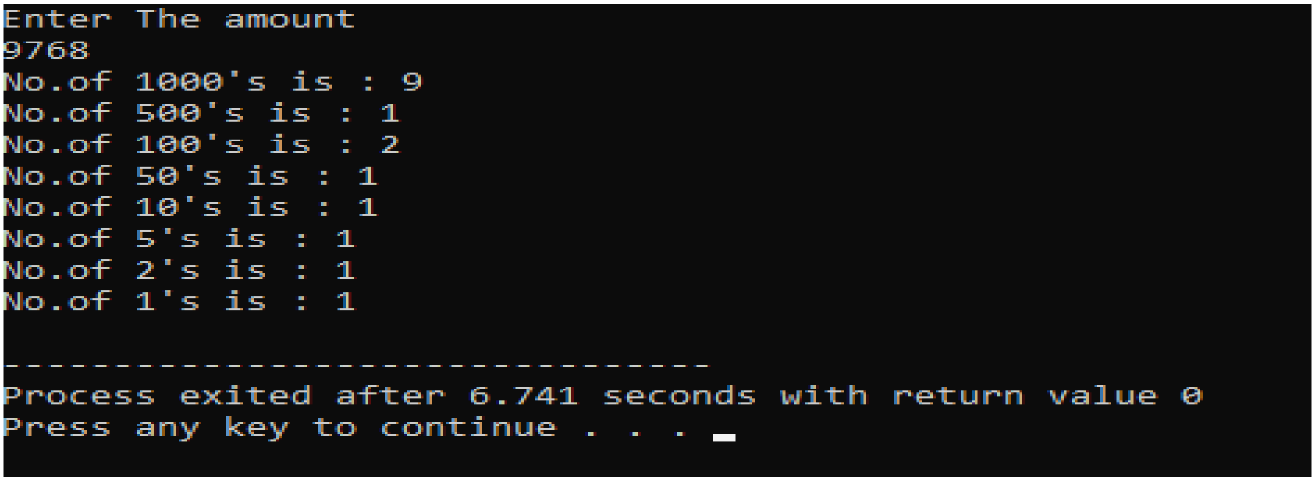
```
Input time in seconds
8659
Hour present: 2
minute present : 24
second present : 18

--------------------------------
Process exited after 3.443 seconds with return value 0
Press any key to continue . . .
```

Task 4: Write a program that prompts for amount in rupees and show how many 1000’s, 500’s, 100’s, 50’s, 10’s, 5’s, 2’s and 1’s in it.

Code:

```
#include<conio.h>
#include<iostream>
#include<string.h>
#include<math.h>
using namespace std;
int main()
 {
   int a;
   cout<<"Enter The amount"<<endl;
   cin>>a;
int th=0;
int fh=0;
int hu=0;
int fif=0;
int te=0;
int fiv=0;
int tw=0;
int on=0;
   while (a>=1000)
   {
      a=a-1000;
      th+=1;
   }
   while (a>=500 && a!=0)
   {
     a=a-500;
      fh+=1;
   }
   while (a>=100 && a!=0)
   {
      a=a-100;
      hu+=1;
   }
   while (a>=50 && a!=0)
   {
      a=a-50;
      fif+=1;}
   while (a>=10 && a!=0)
   {
      a=a-10;
      te+=1;}
   while (a>=5 && a!=0)
  {
      a=a-5;
      fiv+=1;}
   while (a>=2 && a!=0)
  {
      a=a-2;
      tw+=1;}
   while (a>=1 && a>0)
  {
      a=a-1;
      on+=1;
   }
    cout<<"No.of 1000's is : "<<th<<endl;
    cout<<"No.of 500's is : "<<fh<<endl;
    cout<<"No.of 100's is : "<<hu<<endl;
    cout<<"No.of 50's is : "<<fif<<endl;
    cout<<"No.of 10's is : "<<te<<endl;
    cout<<"No.of 5's is : "<<fiv<<endl;
    cout<<"No.of 2's is : "<<tw<<endl;
    cout<<"No.of 1's is : "<<on<<endl;
  return 0;
}
```

Output:



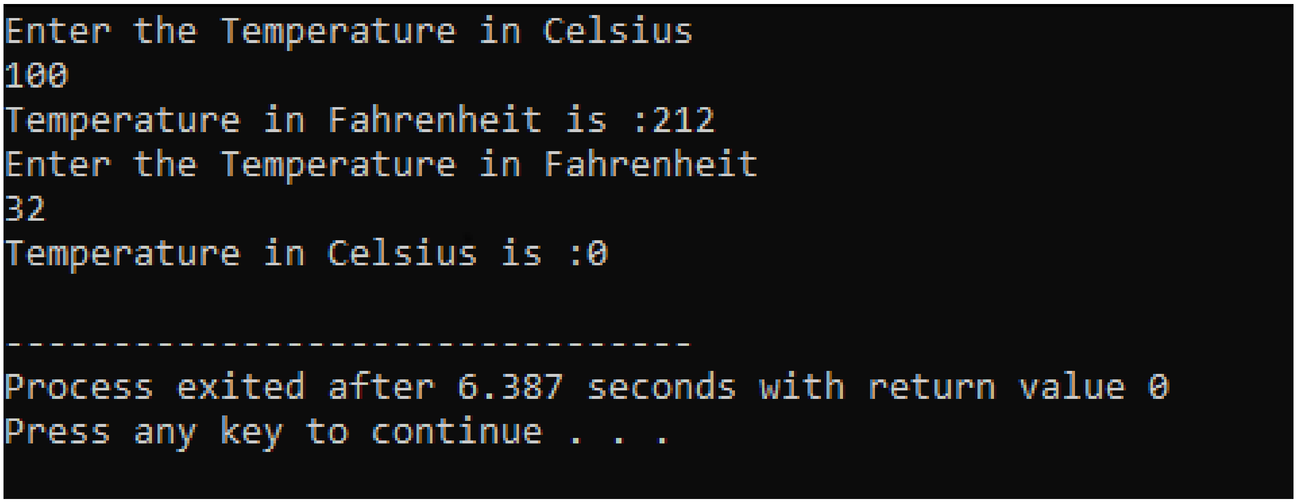
```
Enter The amount
9768
No.of 1000's is : 9
No.of 500's is : 1
No.of 100's is : 2
No.of 50's is : 1
No.of 10's is : 1
No.of 5's is : 1
No.of 2's is : 1
No.of 1's is : 1

--------------------------------
Process exited after 6.741 seconds with return value 0
Press any key to continue . . .
```

Task 5: Write a program that calculates the temperature in Fahrenheit. For that it prompts for temperature in Celsius degrees.Once if the task done do the vice versa.

Code:

```
#include<conio.h>
#include<iostream>
#include<string.h>
#include<math.h>
using namespace std;
ctof(double c1)
{
   double f1 = (c1*(9.0/5.0))+32;
   return f1;
}
ftoc(double f2)
{
   double c2 = (5.0/9.0)*(f2-32);
   return c2;
}
int main()
{
      double fah;
      double cel;
      cout<<"Enter the Temperature in Celsius "<<endl;
      cin>>cel;
      cout<<"Temperature in Fahrenheit is :"<<ctof(cel)<<endl;
      cout<<"Enter the Temperature in Fahrenheit "<<endl;
      cin>>fah;
      cout<<"Temperature in Celsius is :"<<ftoc(fah)<<endl;
          return 0 ;
}
```

Output:



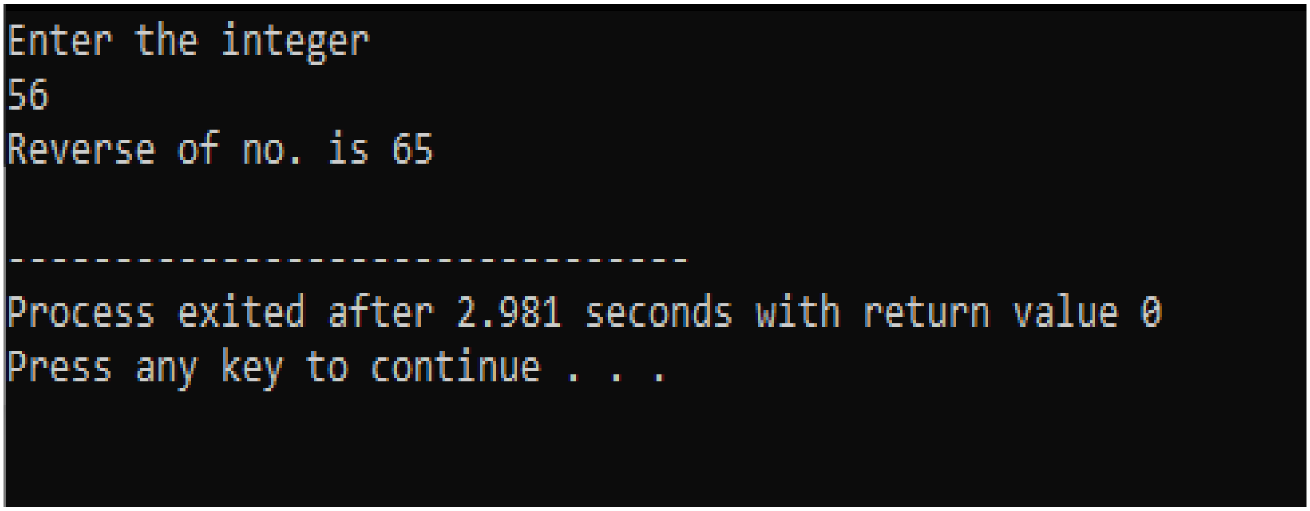
```
Enter the Temperature in Celsius
100
Temperature in Fahrenheit is :212
Enter the Temperature in Fahrenheit
32
Temperature in Celsius is :0

--------------------------------
Process exited after 6.387 seconds with return value 0
Press any key to continue . . .
```

Task 6: Write a program that inputs a two digits integer value, and output its reverse order.

Code:

```
#include<conio.h>
#include <iostream>
#include<bits/stdc++.h>
#include<string.h>

using namespace std;
int reverse(int num)
{
static int rnum = 0;
static int base = 1;
if(num > 0)
{
   reverse(num/10);
   rnum += (num%10)*base;
   base *= 10;
}
return rnum;
}
int main()
{
   int num;
          cout<<"Enter the integer"<<endl;
          cin>>num;
   cout << "Reverse of no. is "<< reverse(num)<<endl;
   return 0;
}
```

Output:



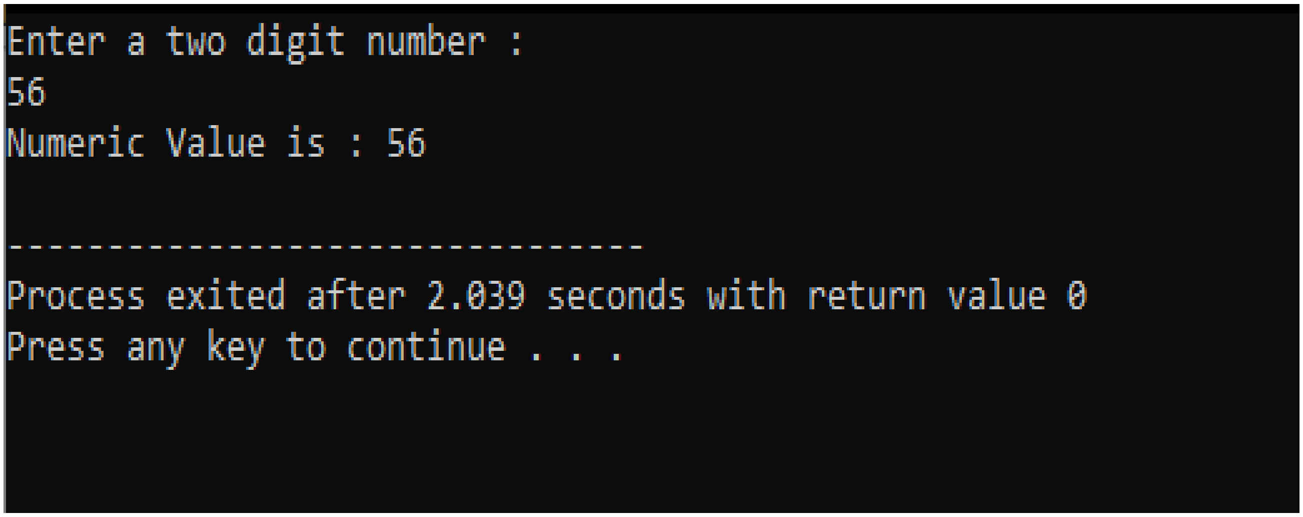
```
Enter the integer
56
Reverse of no. is 65


--------------------------------
Process exited after 2.981 seconds with return value 0
Press any key to continue . . .
```

Task 7: Write a program that reads the two digit nmber as two characters chTen and chUnit and convert that two digit number into an integer value. In order to compute the corresponding integer value, each character must be converted to the digit in the range 0 to 9. this is done by subtracting 48(‘0’) from the ASCII value of the character.
ValueTen=chTen-‘0’; // ’8’-‘0’ is 8
ValueUnit=chUnit-‘0’; // ’2’-‘0’ is 2
To create integer value fro m, the positional value of each digit must be used. In this case multiply ValueTen by 10.
M=ValueTen*10+ValueUnit;//m=8*10+2=82

Code:

```
#include<conio.h>
#include <iostream>
#include<bits/stdc++.h>
#include<string.h>
using namespace std;

int main() {
  string num;
  cout<<"Enter a two digit number : "<<endl;
  cin>>num;
  char chTen ='num[0]';
  char chUnit ='num[1]';
  int Valueten =chTen - '0';
  int Valueunit=chUnit-'0';
 int M=Valueten*10 + Valueunit;
  cout<<"Numeric Value is : "<<M<<endl;
}
```

Output:



```
Enter a two digit number :
56
Numeric Value is : 56


--------------------------------
Process exited after 2.039 seconds with return value 0
Press any key to continue . . .
```

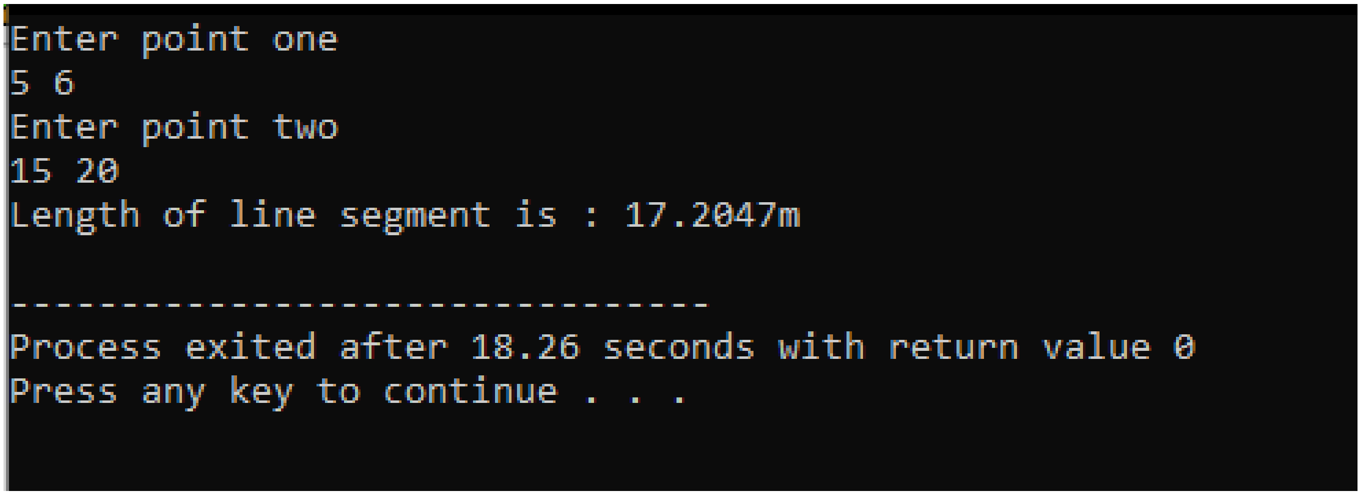
LAB TASK No. 2

Task 1: Write a program to compute the length of the line segment connecting two points.

Code:

```
#include <iostream>
#include <conio.h>
#include <string.h>
#include <math.h>
#include <cmath>
using namespace std;
int main(){
 int x1,y1;
 int x2,y2;
cout<<"Enter point one"<<endl;
cin>>x1>>y1;
cout<<"Enter point two"<<endl;
cin>>x2>>y2;
double distance = sqrt(pow(x2-x1,2)+pow(y2-y1,2));
cout<<"Length of line segment is : "<<distance<<"m"<<endl;
return 0;
}
```

Output:



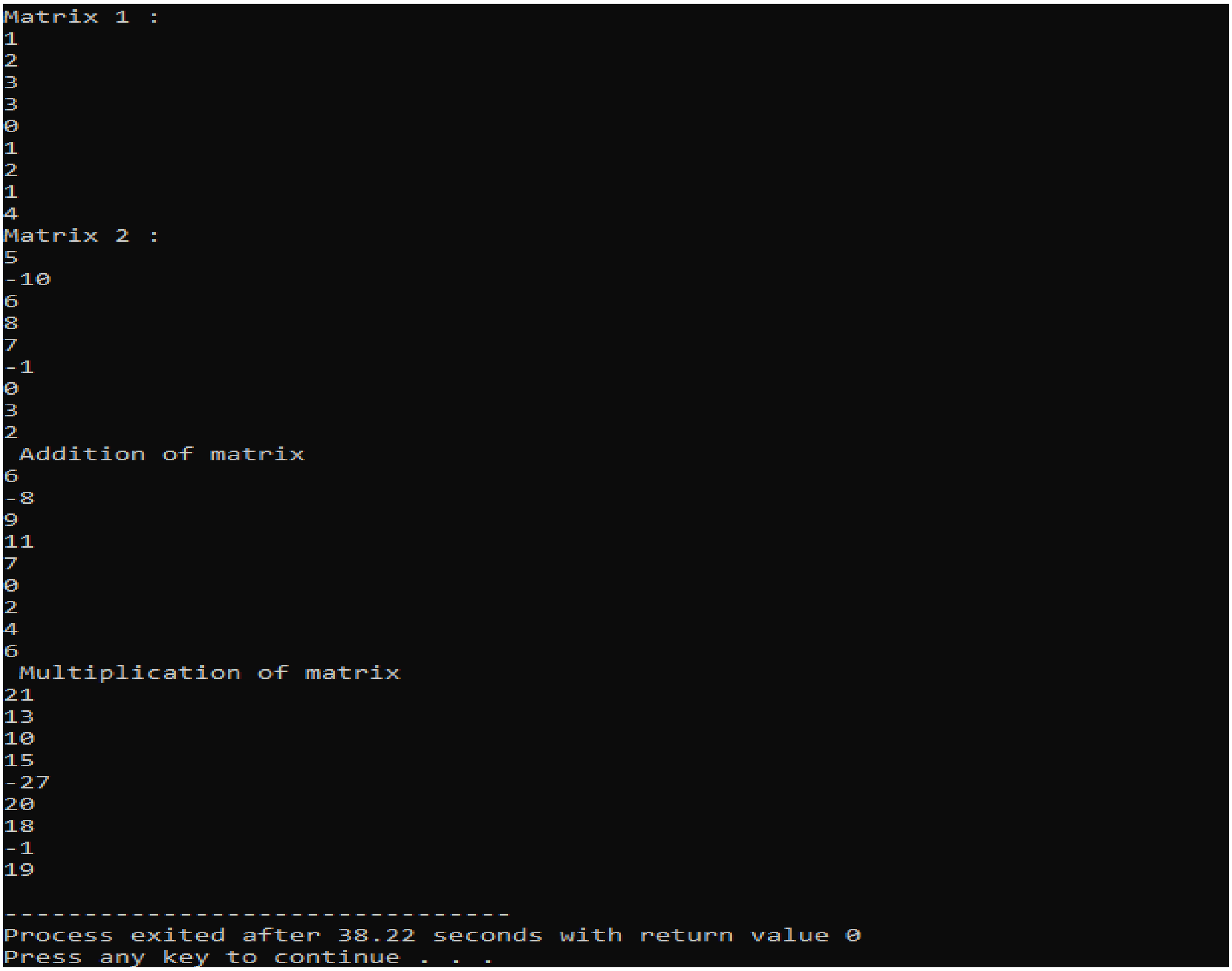
```
Enter point one
5 6
Enter point two
15 20
Length of line segment is : 17.2047m

--------------------------------
Process exited after 18.26 seconds with return value 0
Press any key to continue . . .
```

Task 2: Take two matrix M1 and M2 as an input and perform the addition and multiplication of these two matrices.

Code:

```
#include <iostream>
#include <conio.h>
#include <string.h>
#include <math.h>
#include <cmath>
using namespace std;
int M1[3][3];
int M2[3][3];
int M3[3][3];
int M4[3][3];
int main(){
cout<<"Matrix 1 : "<<endl;
for (int i =0 ; i<3; i++){
        for (int j =0;j<3;j++){
                cin>>M1[i][j];
        }
}
cout<<"Matrix 2 : "<<endl;
for (int i =0 ; i<3; i++){
        for (int j =0;j<3;j++){
                cin>>M2[i][j];
        }
}
cout << " Addition of matrix"<<endl;
for (int i =0 ; i<3; i++){
        for (int j =0;j<3;j++){
                M3[i][j]=M1[i][j]+M2[i][j];
                cout<<M3[i][j]<<endl;
        }
}
cout << " Multiplication of matrix "<<endl;
for (int i =0 ; i<3; i++)
{
        for (int j =0;j<3;j++)
        {
                for (int k=0;k<3;k++)
                {
                M4[i][j]+=M1[i][k]*M2[k][j];
        }
  cout<<M4[i][j]<<endl;
  }
}
return 0;}
```

Output:



```
Matrix 1 :
1
2
3
3
0
1
2
1
4
Matrix 2 :
5
-10
6
8
7
-1
0
3
2
 Addition of matrix
6
-8
9
11
7
0
2
4
6
 Multiplication of matrix
21
13
10
15
-27
20
18
-1
19

--------------------------------
Process exited after 38.22 seconds with return value 0
Press any key to continue . . .
```

Task 3: Write a program that define structure to maintain student records, structure student should be consisting of the following attributes.

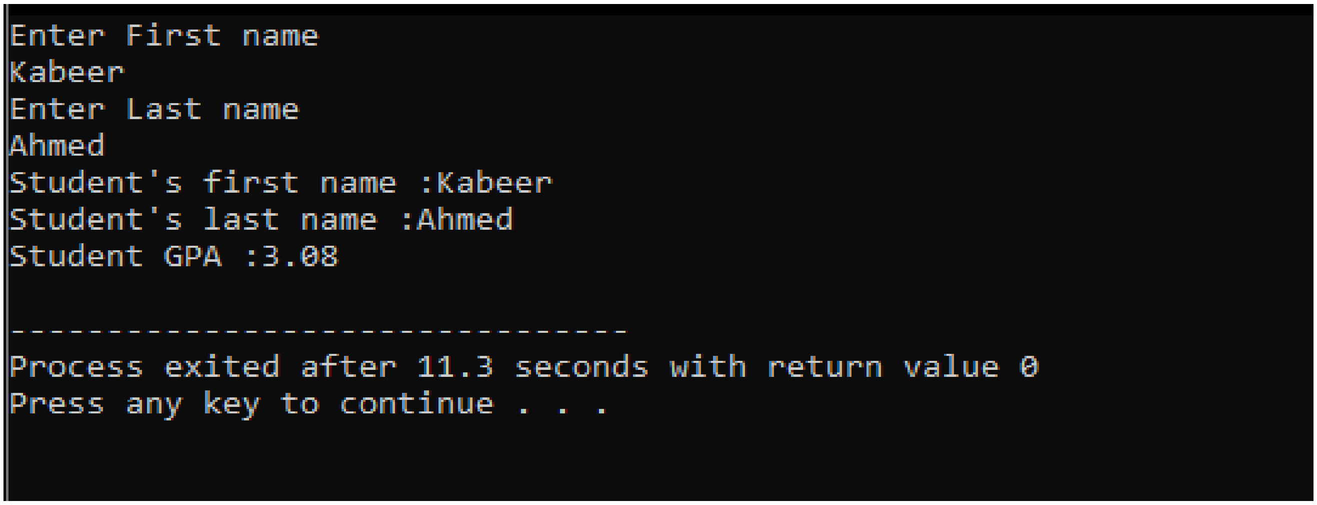
1. Student first name (max 20 characters) 2. Student last name (max 20 characters)

3. Student scores (float/double) e.g 85.4.

Code:

```
#include <iostream>
#include <conio.h>
#include <string.h>
#include <math.h>
#include <cmath>
using namespace std;
struct student
{
        char fname[20] ;
        char lname[20];
        double gpa;
};
int main()
{
student st;
cout<<"Enter First name"<<endl;
cin.get(st.fname,20);
cin.get();
cout<<"Enter Last name"<<endl;
cin.get(st.lname,20);
st.gpa = 3.08;
cout<<"Student's first name :"<<st.fname<<endl;
cout<<"Student's last name :"<<st.lname<<endl;
cout<<"Student GPA :"<<st.gpa<<endl;
return 0;
}
```

Output:



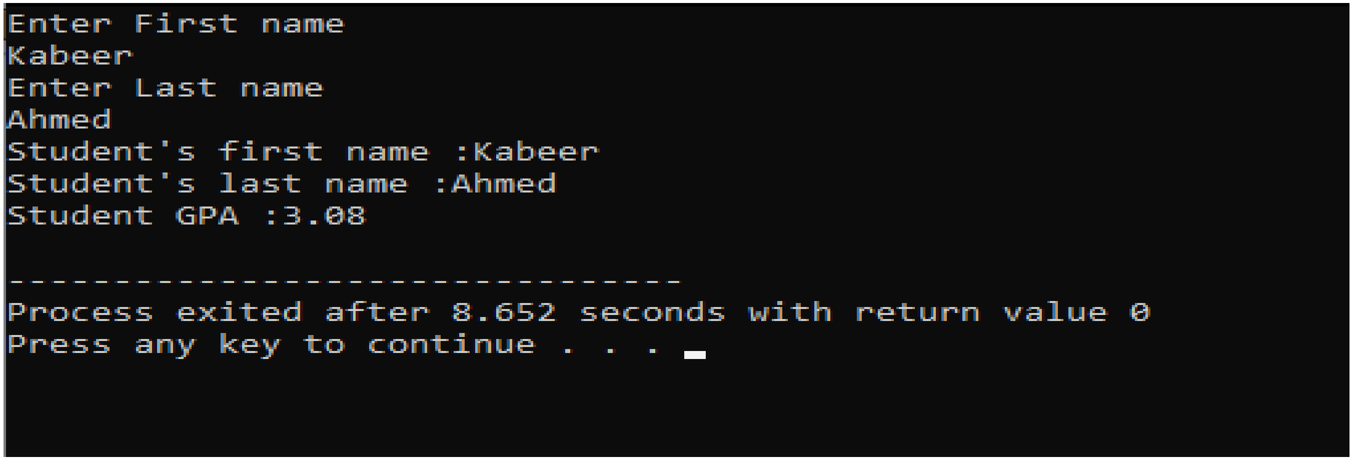
```
Enter First name
Kabeer
Enter Last name
Ahmed
Student's first name :Kabeer
Student's last name :Ahmed
Student GPA :3.08

--------------------------------
Process exited after 11.3 seconds with return value 0
Press any key to continue . . .
```

Task 4: Pass the structure define in Q.3 to some function to move to display.

Code:

```
#include <iostream>
#include <conio.h>
#include <string.h>
#include <math.h>
#include <cmath>
using namespace std;
struct student
{
	char fname[20] ;
	char lname[20];
	double gpa;
};
void showdata(student s)
{
cout<<"Enter First name"<<endl;
cin.get(s.fname,20);
cin.get();
cout<<"Enter Last name"<<endl;
cin.get(s.lname,20);
s.gpa = 3.08;
cout<<"Student's first name :"<<s.fname<<endl;
cout<<"Student's last name :"<<s.lname<<endl;
cout<<"Student GPA :"<<s.gpa<<endl;

}
int main()
{
student st;
showdata(st);
return 0;
}
```

Output:



```
Enter First name
Kabeer
Enter Last name
Ahmed
Student's first name :Kabeer
Student's last name :Ahmed
Student GPA :3.08

--------------------------------
Process exited after 8.652 seconds with return value 0
Press any key to continue . . .
```

Task 5: Create nested structure . Firstly define Address structure and then call address Structure in Employee Structure and program will give some raise in salary ,if it is less than 50000

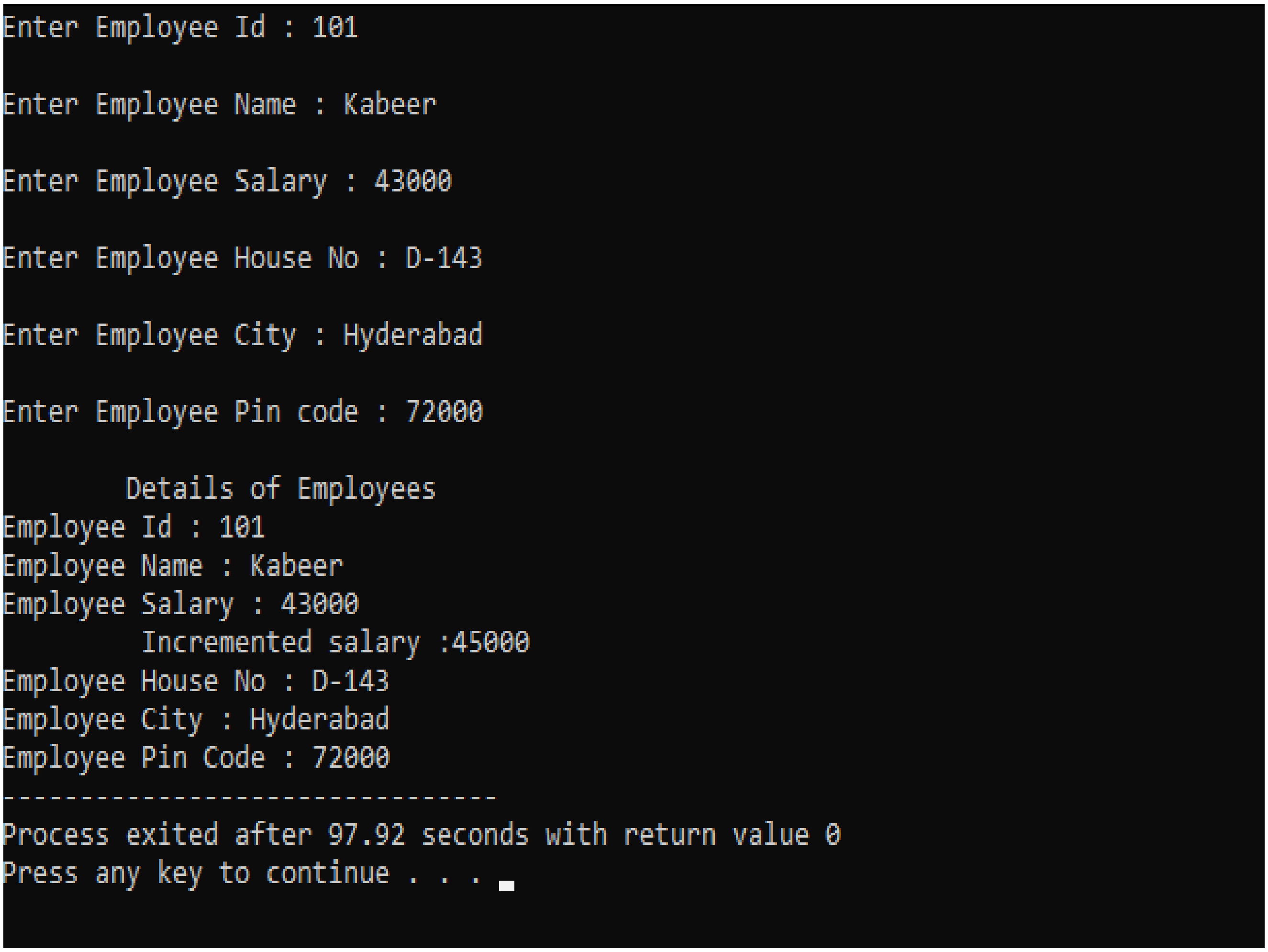
Address (house no, city, pin code)

Employee (empid,name,salary,address)

Code:

```
#include <iostream>
#include <conio.h>
#include <string.h>
#include <math.h>
#include <cmath>
using namespace std;
struct Address
{
   char HouseNo[20];
   char City[20];
   char PinCode[20];
};
struct Employee
{
   int Id;
   char Name[15];
   float Salary;
   struct Address Add;
};
int increment(Employee sl)
{
   sl.Salary+=2000;
   return sl.Salary;
}
int main()
{
int i;
Employee Emp;
cout << "Enter Employee Id : ";
cin >> Emp.Id;
cout << "\nEnter Employee Name : ";
cin >> Emp.Name;
cout << "\nEnter Employee Salary : ";
cin >> Emp.Salary;
cout << "\nEnter Employee House No : ";
cin >> Emp.Add.HouseNo;
cout << "\nEnter Employee City : ";
cin >> Emp.Add.City;
cout << "\nEnter Employee Pin code : ";
cin >> Emp.Add.PinCode;
cout << "\n\tDetails of Employees";
cout << "\nEmployee Id : " << Emp.Id;
cout << "\nEmployee Name : " << Emp.Name;
cout << "\nEmployee Salary : " << Emp.Salary;
if(Emp.Salary<=50000)
{
   cout<<"\n\t Incremented salary :"<<increment(Emp);
}
cout << "\nEmployee House No : " << Emp.Add.HouseNo;
cout << "\nEmployee City : " << Emp.Add.City;
cout << "\nEmployee Pin Code : " << Emp.Add.PinCode;

return 0;
}
```

Output:



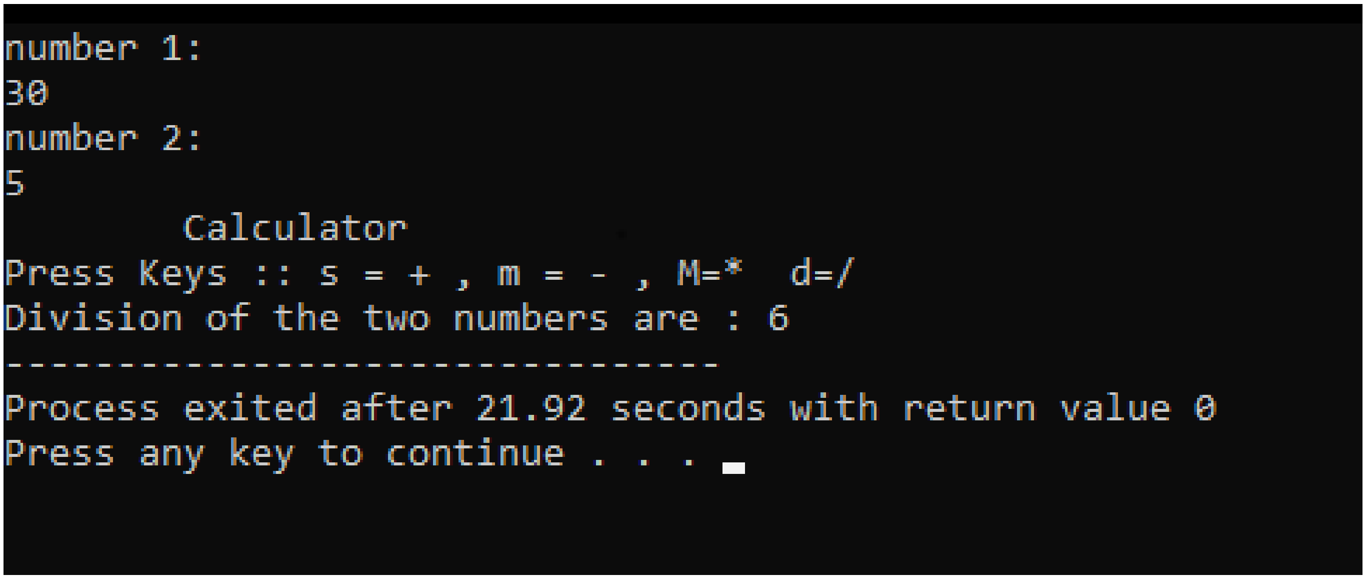
```
Enter Employee Id : 101

Enter Employee Name : Kabeer

Enter Employee Salary : 43000

Enter Employee House No : D-143

Enter Employee City : Hyderabad

Enter Employee Pin code : 72000

        Details of Employees
Employee Id : 101
Employee Name : Kabeer
Employee Salary : 43000
         Incremented salary :45000
Employee House No : D-143
Employee City : Hyderabad
Employee Pin Code : 72000
--------------------------------
Process exited after 97.92 seconds with return value 0
Press any key to continue . . .
```

Task 6: Write a C++ Program to create a small Calculator by using Pointers.

Code:

```
#include <iostream>
#include <conio.h>
#include <string.h>
#include <math.h>
#include <cmath>
using namespace std;
int add(int *p1,int *p2)
{
	return *p1 + *p2;
}

int sub(int *p1,int *p2)
{
	return *p1 - *p2;
}
int multiply(int *p1,int *p2)
{
	return *p1 * *p2;
}
int divide(int *p1,int *p2)
{
	return *p1 / *p2;
}
int main()
{
int num1 , num2;
cout<<"number 1:"<<endl;
cin>>num1;
cout<<"number 2:"<<endl;
cin >> num2;
int *p1 ,*p2;
p1=&num1;
p2=&num2;
char dir = '0';
cout<<"\tCalculator"<<endl;
cout<<"Press Keys :: s = + , m = - , M=*  d=/ "<<endl;
dir=_getch();
if(dir=='s')
{
   cout<<"Addition of the two numbers are : "<<add(p1,p2);
}
else if(dir=='m')
{
        cout<<"Subtraction of the two numbers are : "<<sub(p1,p2);
}
else if(dir=='M')
{
   cout<<"Multiplication of the two numbers are : "<<multiply(p1,p2);
}
else if(dir=='d')
{
        cout<<"Division of the two numbers are : "<<divide(p1,p2);
}
else
{
        cout<<"Operation is out of range";
}

return 0;
}
```

Output:



```
number 1:
30
number 2:
5
        Calculator
Press Keys :: s = + , m = - , M=*  d=/
Division of the two numbers are : 6
--------------------------------
Process exited after 21.92 seconds with return value 0
Press any key to continue . . .
```

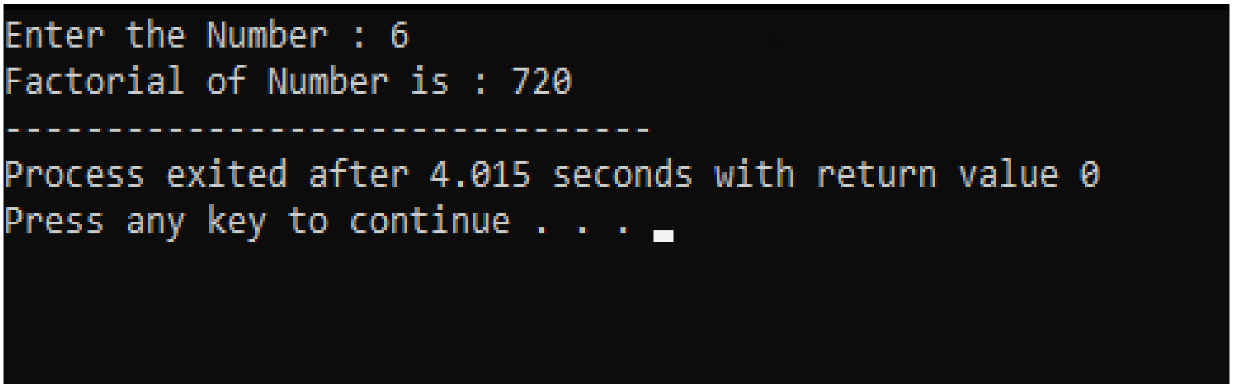
LAB TASK No. 3

Task 1: write a program in C++ that can calculate the factorial of a number by passing the address of that number to a function,using pointers.

Code:

```
#include <iostream>
#include <conio.h>
#include <string.h>
#include <math.h>
#include <cmath>
using namespace std;
int fact(int *p)
{
   int factorial=1;
   while(*p>1)
         {
      factorial*=*p;
      *p=*p-1;
   }
   return factorial;
}
int main()
{
         int num,*p;
         cout<<"Enter the Number : ";
         cin>>num;
         p=&num;
         cout <<"Factorial of Number is : "<<fact(p);
         return 0;
}
```

Output:



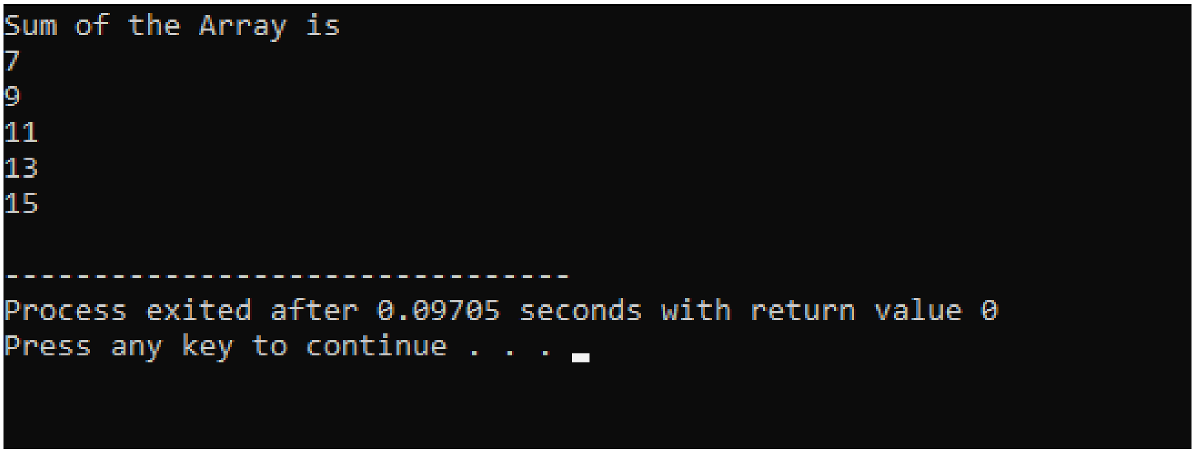
```
Enter the Number : 6
Factorial of Number is : 720
--------------------------------
Process exited after 4.015 seconds with return value 0
Press any key to continue . . .
```

Task 2: Write down a C++ program that will declare and initialize two arrays and would generate the sum of these two arrays by using pointers.

Code:

```
#include <iostream>
#include <conio.h>
#include <string.h>
#include <math.h>
#include <cmath>
using namespace std;
void sum(int *p1,int *p2)
{
        int c[5];
        int *sum =c;
        cout<<"Sum of the Array is"<<endl;
        for (int i=0;i<5;i++)
        {
                *sum = *p1 + *p2;
                cout<<*sum<<endl;
                *p1++;
                *p2++;
   }
}
int main()
{
        int *p1,*p2;
        int arr1[]={1,2,3,4,5};
        int arr2[]={6,7,8,9,10};
        p1=arr1;
        p2=arr2;
        sum(p1,p2);
        return 0;
}
```

Output:



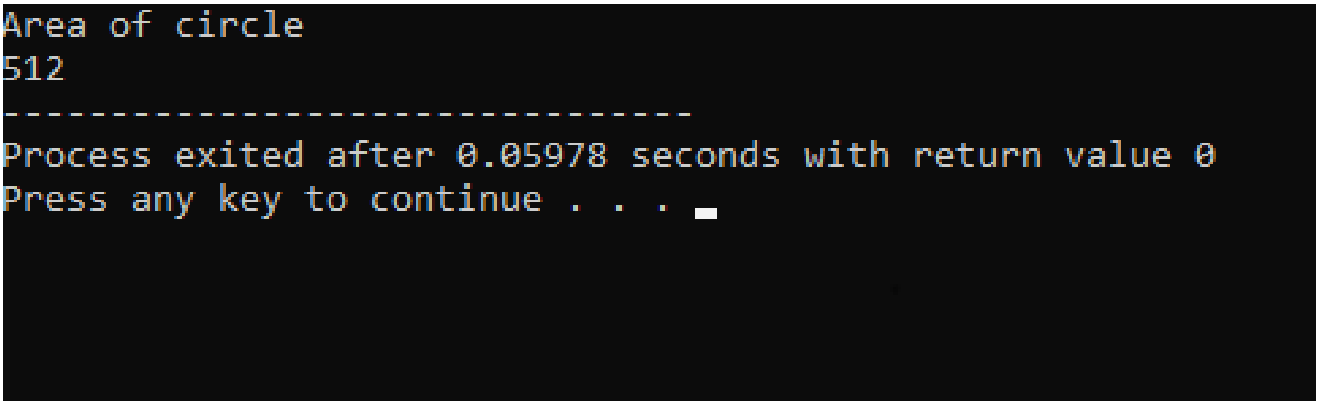
```
Sum of the Array is
7
9
11
13
15

--------------------------------
Process exited after 0.09705 seconds with return value 0
Press any key to continue . . .
```

Task 3: Write down a C++ program, that will Calculate the area of a Circle by using Constant Data member PI=3.14, and by using Constant Pointer.

Code:

```
#include <iostream>
#include <conio.h>
#include <string.h>
#include <math.h>
#include <cmath>
using namespace std;
int main()
{
int radius =8;
int pivalue = 3.14;
const int *const ptr2 = &radius;
 cout << "Area of circle "<<endl;
 cout<< *ptr2 * pow(radius,2);
return 0;
}
```

Output:



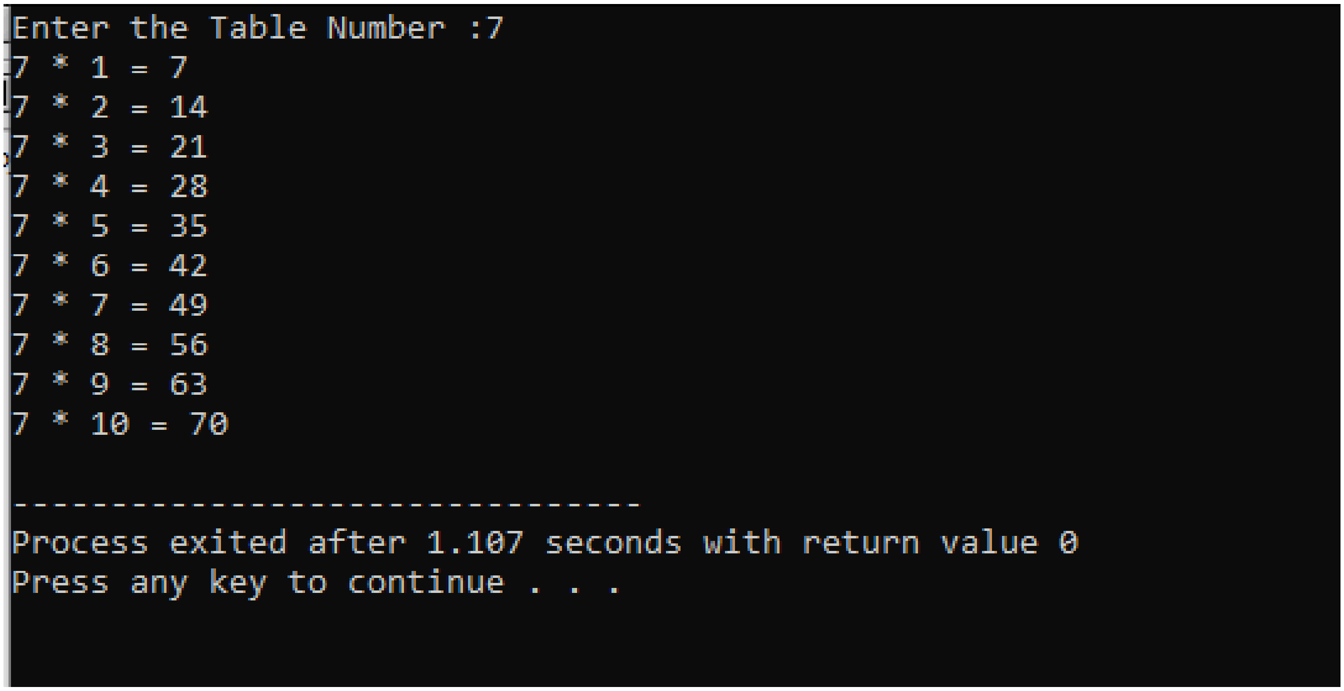
```
Area of circle
512
--------------------------------
Process exited after 0.05978 seconds with return value 0
Press any key to continue . . .
```

Task 4: Write down a C++ program that would generate some table 2*1=2,2*10=20by using pointers.

Code:

```
#include <iostream>
#include <conio.h>
#include <string.h>
#include <math.h>
#include <cmath>
using namespace std;
int main()
{
int num;
	int *p;
	cout<<"Enter the Table Number :";
	cin>>num;
	cin.get();
	p=&num;
	for (int i =1 ; i<11; i++)
	{
	   cout<< *p <<" * "<<i<<" = "<<*p * i<<endl;
	}
	return 0;
}
```

Output:



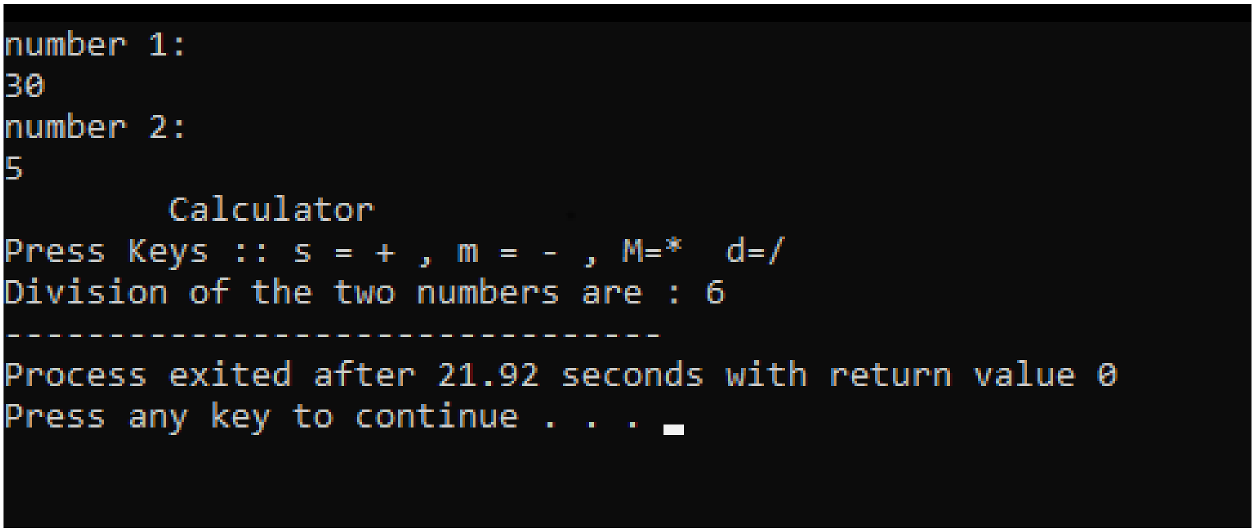
```
Enter the Table Number :7
7 * 1 = 7
7 * 2 = 14
7 * 3 = 21
7 * 4 = 28
7 * 5 = 35
7 * 6 = 42
7 * 7 = 49
7 * 8 = 56
7 * 9 = 63
7 * 10 = 70


--------------------------------
Process exited after 1.107 seconds with return value 0
Press any key to continue . . .
```

Task 5: Write down a C++ program that would design a simple Calculator, by using Pointers.

Code:

```
#include <iostream>
#include <conio.h>
#include <string.h>
#include <math.h>
#include <cmath>
using namespace std;
int add(int *p1,int *p2)
{
	return *p1 + *p2;
}

int sub(int *p1,int *p2)
{
	return *p1 - *p2;
}
int multiply(int *p1,int *p2)
{
	return *p1 * *p2;
}
int divide(int *p1,int *p2)
{
	return *p1 / *p2;
}
int main()
{
int num1 , num2;
cout<<"number 1:"<<endl;
cin>>num1;
cout<<"number 2:"<<endl;
cin >> num2;
int *p1 ,*p2;
p1=&num1;
p2=&num2;
char dir = '0';
cout<<"\tCalculator"<<endl;
cout<<"Press Keys :: s = + , m = - , M=*  d=/ "<<endl;
dir=_getch();
if(dir=='s')
{
   cout<<"Addition of the two numbers are : "<<add(p1,p2);
}
else if(dir=='m')
{
        cout<<"Subtraction of the two numbers are : "<<sub(p1,p2);
}
else if(dir=='M')
{
   cout<<"Multiplication of the two numbers are : "<<multiply(p1,p2);
}
else if(dir=='d')
{
        cout<<"Division of the two numbers are : "<<divide(p1,p2);
}
else
{
        cout<<"Operation is out of range";
}

return 0;
}
```

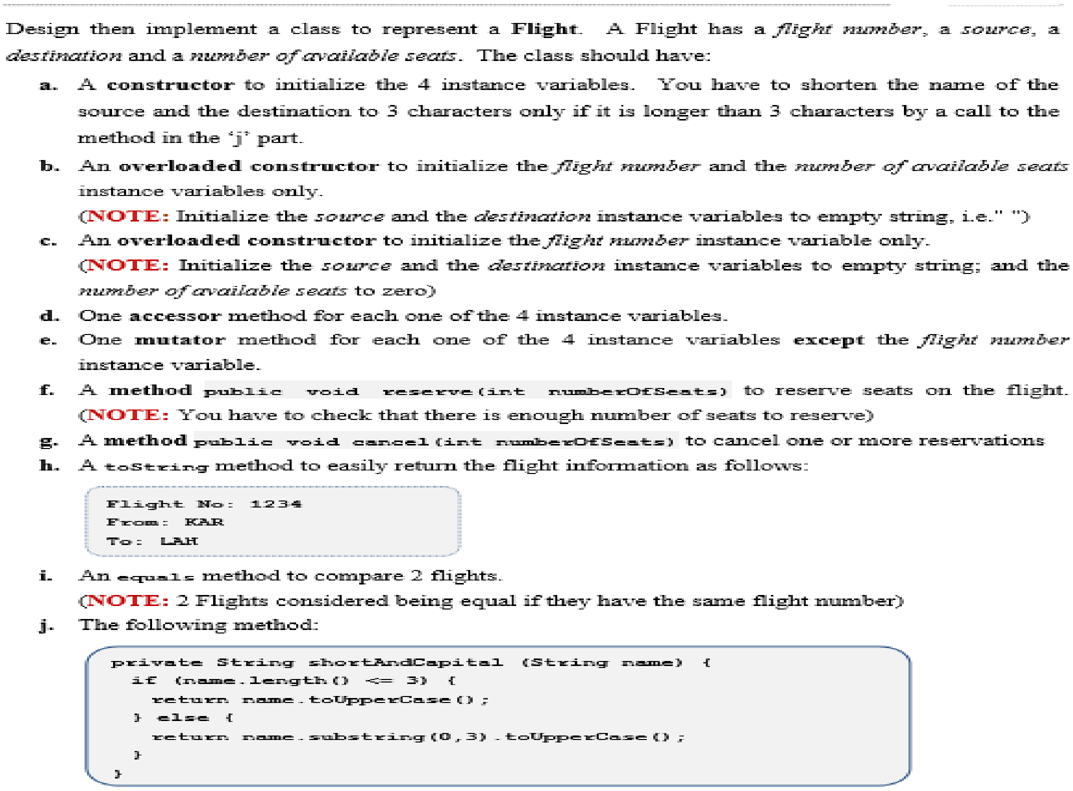
Output:



```
number 1:
30
number 2:
5
        Calculator
Press Keys :: s = + , m = - , M=*  d=/
Division of the two numbers are : 6
--------------------------------
Process exited after 21.92 seconds with return value 0
Press any key to continue . . .
```

LAB TASK No. 4

Task 1: Create the object of the flight class and apply the defined methods.



Design then implement a class to represent a **Flight**. A Flight has a *flight number*, a *source*, a *destination* and a *number of available seats*. The class should have:

a. A **constructor** to initialize the 4 instance variables. You have to shorten the name of the source and the destination to 3 characters only if it is longer than 3 characters by a call to the method in the 'j' part.

b. An **overloaded constructor** to initialize the *flight number* and the *number of available seats* instance variables only.
(**NOTE:** Initialize the *source* and the *destination* instance variables to empty string, i.e." ")

c. An **overloaded constructor** to initialize the *flight number* instance variable only.
(**NOTE:** Initialize the *source* and the *destination* instance variables to empty string; and the *number of available seats* to zero)

d. One **accessor** method for each one of the 4 instance variables.

e. One **mutator** method for each one of the 4 instance variables **except** the *flight number* instance variable.

f. A **method** `public void reserve(int numberOfSeats)` to reserve seats on the flight.
(**NOTE:** You have to check that there is enough number of seats to reserve)

g. A **method** `public void cancel(int numberOfSeats)` to cancel one or more reservations

h. A `toString` method to easily return the flight information as follows:

```
Flight No: 1234
From: KAR
To: LAH
```

i. An `equals` method to compare 2 flights.
(**NOTE:** 2 Flights considered being equal if they have the same flight number)

j. The following method:

```
private String shortAndCapital (String name) {
  if (name.length() <= 3) {
    return name.toUpperCase();
  } else {
    return name.substring(0,3).toUpperCase();
  }
}
```

Code:

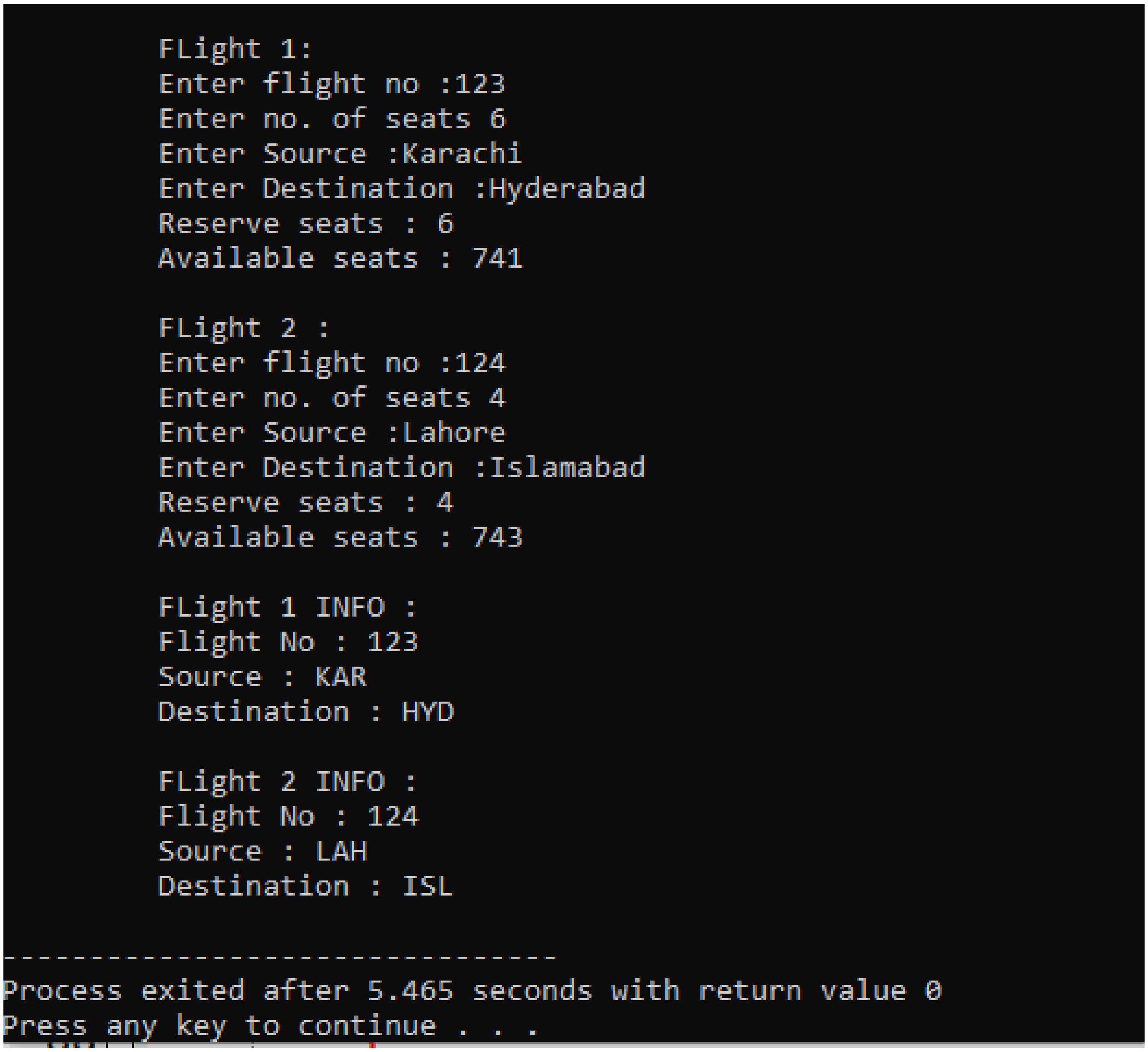
```
#include <iostream>
#include <conio.h>
#include <string>
#include <math.h>
#include <cmath>
#include <iostream>
#include <string>
using namespace std;
class Flight
{
int cance;
int reserv;
string source;
string destination;
int avaSeats;
int flightNo;
public:
Flight(string s,string d,int ava , int fNo): source(s),destination(d),avaSeats(ava),flightNo(fNo){  }
Flight(int ava , int fNo)
```

```
{
 source=" ";
 destination=" ";
 avaSeats=ava;
 flightNo=fNo;
}
Flight(int fNo)
{
 source=" ";
 destination=" ";
 avaSeats=0;
 flightNo=fNo;
}
void setSource(string s)
{
	source=s;
}
void setDestination(string d)
{
	destination=d;
}
void setSeats(int a)
{
	avaSeats=a;
}
void setflight(int f)
{
	flightNo=f;
}
string getSource()
{
	return source;
}
string getDestination()
{
	return destination;
}
int getSeats(){
	return avaSeats;
}
int getflight(){
	return flightNo;
}
int reserve(int numberofseat){
	reserv=numberofseat;
	avaSeats-=reserv;
	return avaSeats;
}
int cancel(int numberofseats){
	cance=numberofseats;
	avaSeats+=cance;
```

```
        return cance;
}
void equals(int f, int f1){
if (f==f1){
        cout<< "\n\tequal flights";
}
}

void tostring(string s,string d, int fNo){
        cout<<"\n\t Flight No : "<< fNo;
        for(int i=0;i<=3;i++)
        {
                if(s[i]>=97 && s[i]<=122)
     {
                   s[i]=s[i]-32;
     }
     if(d[i]>=97 && d[i]<=122)
     {
                   d[i]=d[i]-32;
     }
  }
        cout<<"\n\t Source : "<<s[0]<<s[1]<<s[2];
        cout<<"\n\t Destination : "<<d[0]<<d[1]<<d[2]<<endl;
}
};

int main()
{
Flight f1("Karachi","Hyderabad",747,123);
Flight f2("Lahore","Islamabad",747,124);
int f1fl= f1.getflight();
string f1s= f1.getSource();
string f1d=f1.getDestination();

int f2fl= f2.getflight();
string f2s= f2.getSource();
string f2d=f2.getDestination();
cout<<"\n\t FLight 1:";
cout<<"\n\t Enter flight no :"<<f1.getflight();
int nseats,nseats1;
cout<<"\n\t Enter no. of seats ";
cin>>nseats;
cout<<"\t Enter Source :"<<f1.getSource();
cout<<"\n\t Enter Destination :"<<f1.getDestination();
cout<<"\n\t Reserve seats : "<< nseats;
cout<<"\n\t Available seats : "<< f1.reserve(nseats)<<endl;

cout<<"\n\t FLight 2 :";
cout<<"\n\t Enter flight no :"<<f2.getflight();

cout<<"\n\t Enter no. of seats ";
```

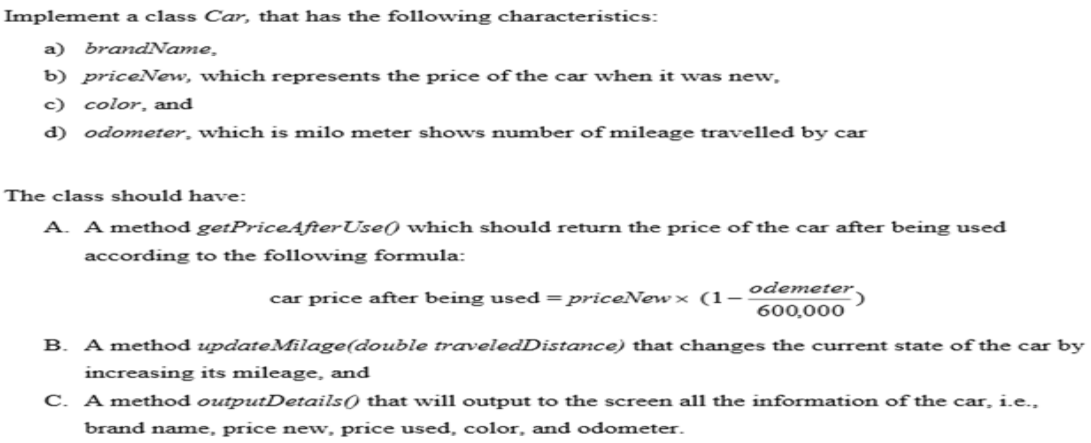
```
cin>>nseats1;
cout<<"\t Enter Source :"<<f2.getSource();
cout<<"\n\t Enter Destination :"<<f2.getDestination();
cout<<"\n\t Reserve seats : "<< nseats1;
cout<<"\n\t Available seats : "<< f2.reserve(nseats1)<<endl;

cout<<"\n\t FLight 1 INFO :";
f1.tostring(f1s,f1d,f1fl);

cout<<"\n\t FLight 2 INFO :";
f2.tostring(f2s,f2d,f2fl);
f1.equals(f1fl,f2fl);

        return 0;
}
```

Output:



```
        FLight 1:
        Enter flight no :123
        Enter no. of seats 6
        Enter Source :Karachi
        Enter Destination :Hyderabad
        Reserve seats : 6
        Available seats : 741

        FLight 2 :
        Enter flight no :124
        Enter no. of seats 4
        Enter Source :Lahore
        Enter Destination :Islamabad
        Reserve seats : 4
        Available seats : 743

        FLight 1 INFO :
        Flight No : 123
        Source : KAR
        Destination : HYD

        FLight 2 INFO :
        Flight No : 124
        Source : LAH
        Destination : ISL

--------------------------------
Process exited after 5.465 seconds with return value 0
Press any key to continue . . .
```

Task 2:



Implement a class *Car,* that has the following characteristics:

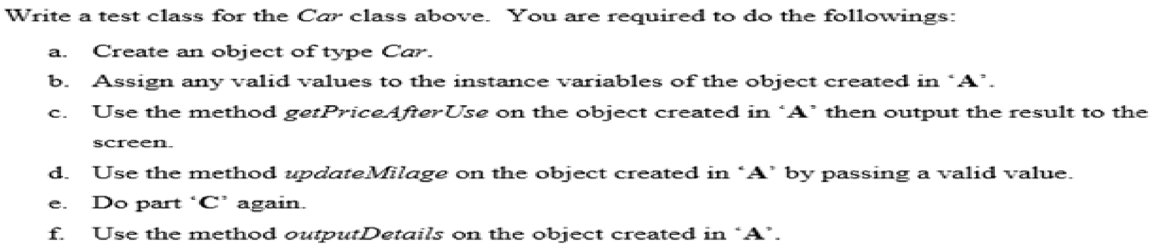
a) *brandName,*
b) *priceNew,* which represents the price of the car when it was new,
c) *color,* and
d) *odometer,* which is milo meter shows number of mileage travelled by car

The class should have:

A. A method *getPriceAfterUse()* which should return the price of the car after being used according to the following formula:

$$\text{car price after being used} = priceNew \times \left(1 - \frac{odemeter}{600{,}000}\right)$$

B. A method *updateMilage(double traveledDistance)* that changes the current state of the car by increasing its mileage, and
C. A method *outputDetails()* that will output to the screen all the information of the car, i.e., brand name, price new, price used, color, and odometer.



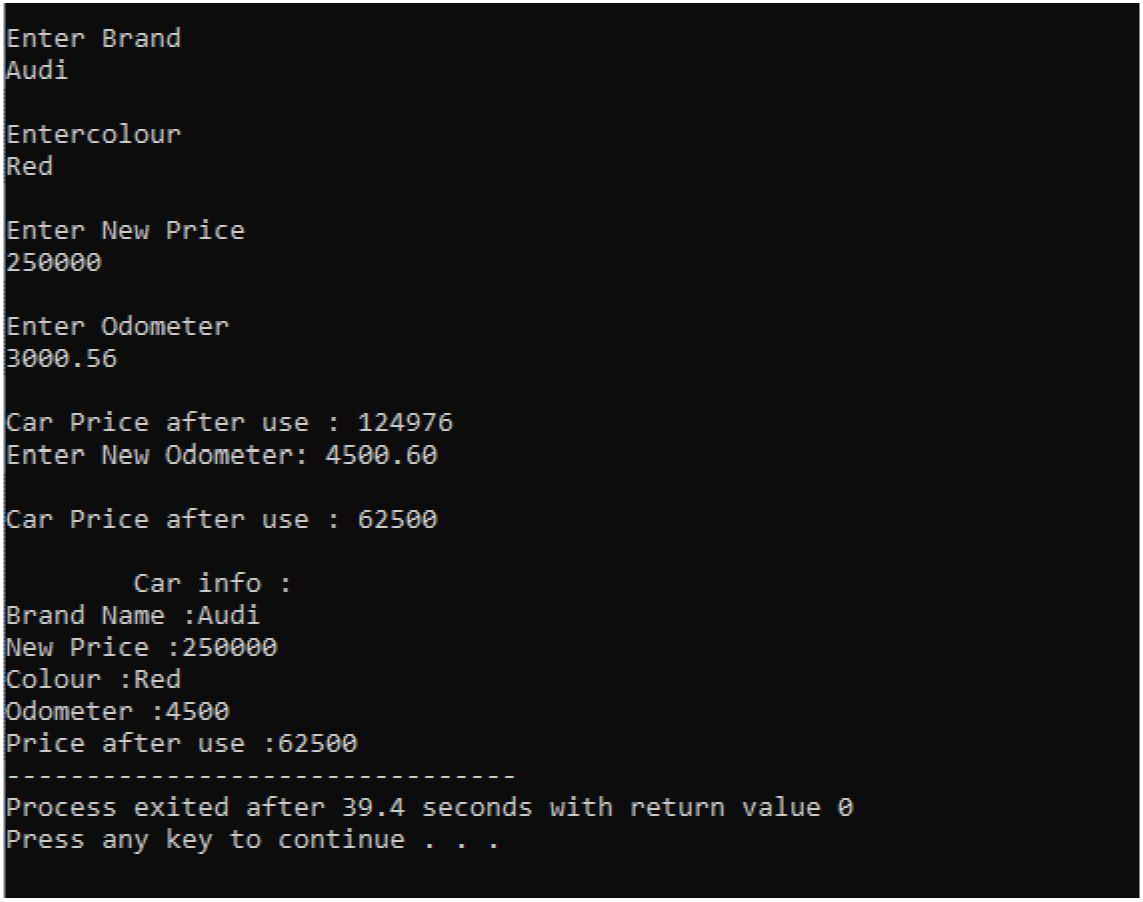
Write a test class for the *Car* class above. You are required to do the followings:

a. Create an object of type *Car.*
b. Assign any valid values to the instance variables of the object created in '**A**'.
c. Use the method *getPriceAfterUse* on the object created in '**A**' then output the result to the screen.
d. Use the method *updateMilage* on the object created in '**A**' by passing a valid value.
e. Do part '**C**' again.
f. Use the method *outputDetails* on the object created in '**A**'.

Code:

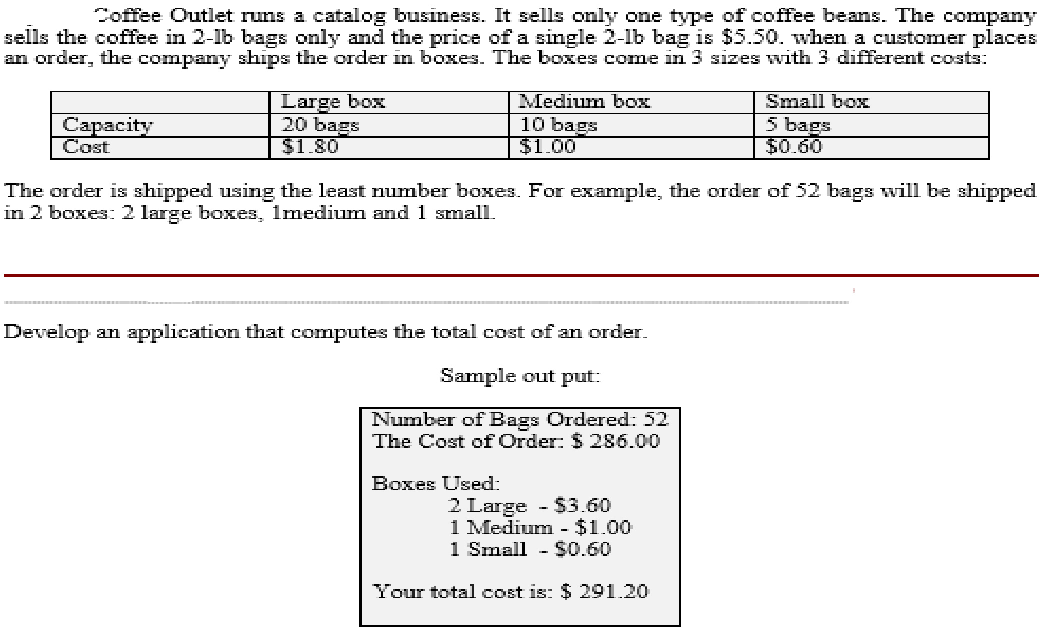
```
#include <iostream>
#include <conio.h>
#include <string>
#include <math.h>
#include <cmath>
using namespace std;
class Car{
        string brandname;
        double priceNew;
        string color;
        double odometer;
        public:
                Car(string b , double p,string c,double o) : brandname(b),
priceNew(p),odometer(o),color(c) {        }

        int getpriceafteruse(double o,double pp){
                double use = pp * (1 - (o/6000));
                return use;
        }
        int updatedodometer(double o){
                odometer+=o;
                return o;
        }
        int outputdetails(string b , double pn, double pu,string c ,double o){
                cout<< "\n\tCar info :";
                cout<< "\nBrand Name :"<<b;
                cout<< "\nNew Price :"<<pn;
                cout<< "\nColour :"<<c;
                cout<< "\nOdometer :"<<o;
                cout<< "\nPrice after use :"<<pu;
        }
};
int main(){
string b;
double p;
string c;
double o;
double oo;
cout<< "\nEnter Brand "<<endl;
cin>>b;
cout<< "\nEntercolour "<<endl;
cin>>c;
cout<< "\nEnter New Price "<<endl;
cin>>p;
cout<< "\nEnter Odometer "<<endl;
cin>>o;
Car cc(b,p,c,o);
cout<< "\nCar Price after use : ";
cout<<cc.getpriceafteruse(o,p);
cout<< "\nEnter New Odometer: ";
cin>>oo;
double up=cc.updatedodometer(oo);
double pu=cc.getpriceafteruse(up,p);
cout<< "\nCar Price after use : "<<cc.getpriceafteruse(up,p)<<endl;
cc.outputdetails(b,p,pu,c,up);
return 0;

}
```

Output:



```
Enter Brand
Audi

Entercolour
Red

Enter New Price
250000

Enter Odometer
3000.56

Car Price after use : 124976
Enter New Odometer: 4500.60

Car Price after use : 62500

        Car info :
Brand Name :Audi
New Price :250000
Colour :Red
Odometer :4500
Price after use :62500
--------------------------------
Process exited after 39.4 seconds with return value 0
Press any key to continue . . .
```

Task 3:



Coffee Outlet runs a catalog business. It sells only one type of coffee beans. The company sells the coffee in 2-lb bags only and the price of a single 2-lb bag is $5.50. when a customer places an order, the company ships the order in boxes. The boxes come in 3 sizes with 3 different costs:

	Large box	Medium box	Small box
Capacity	20 bags	10 bags	5 bags
Cost	$1.80	$1.00	$0.60

The order is shipped using the least number boxes. For example, the order of 52 bags will be shipped in 2 boxes: 2 large boxes, 1medium and 1 small.

Develop an application that computes the total cost of an order.

Sample out put:

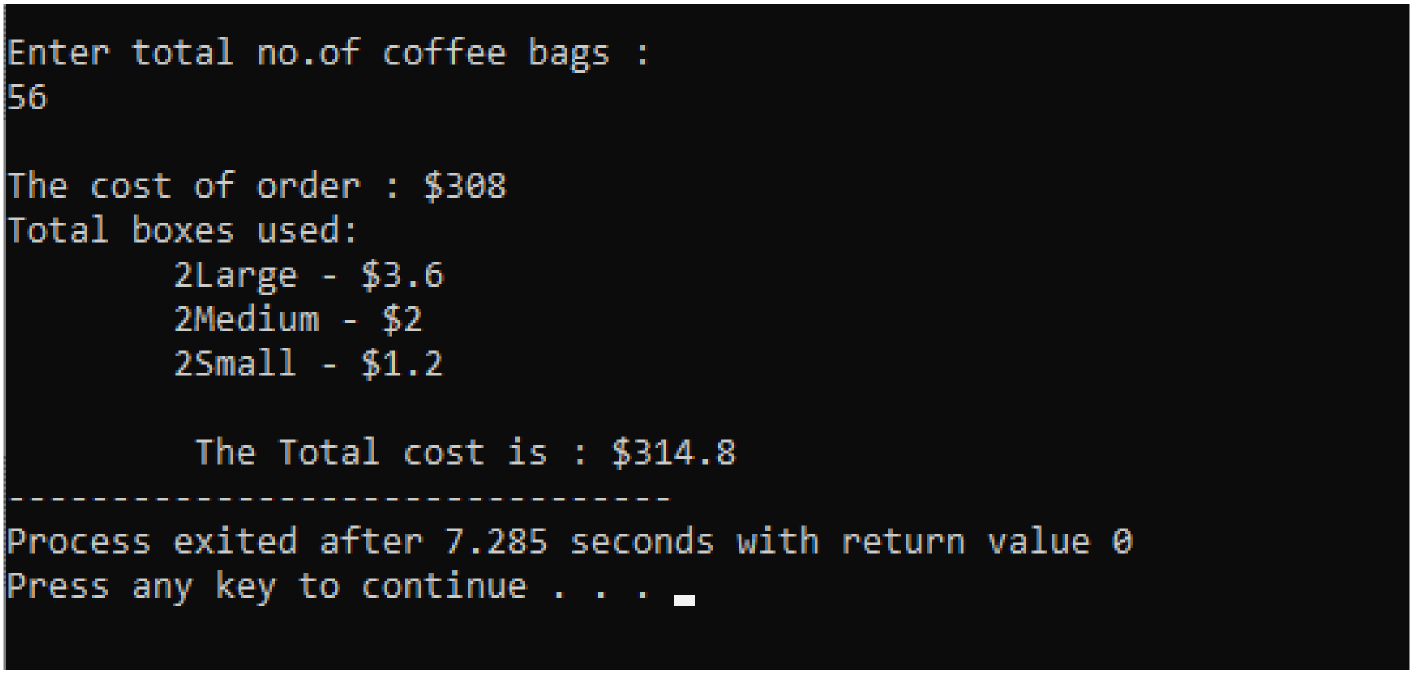
```
Number of Bags Ordered: 52
The Cost of Order: $ 286.00

Boxes Used:
          2 Large  - $3.60
          1 Medium - $1.00
          1 Small  - $0.60

Your total cost is: $ 291.20
```

Code:

```
#include <iostream>
#include <conio.h>
#include <string>
#include <math.h>
#include <cmath>
using namespace std;
class Coffee
{
int total;
int big ;
int medium ;
int small ;
public:
void div(int t)
{
	double tt=t*5.50;
	cout<<"\nThe cost of order : $"<<tt;
	total=t;
	cout<<"\nTotal boxes used: "<<endl;
	while(total>=20)
	{
		big+=1;
		total-=20;
	}
```

```
		double bigt = big*1.80;
		cout<<"\t"<<big<<"Large - $"<<bigt<<endl;
		while(total>=10 && total<20)
		{
			medium+=1;
			total-=10;
		}
		double mt = medium*1.00;
		cout<<"\t"<<medium<<"Medium - $"<<mt<<endl;
		while(total>=5 || (total<=5 && total>0))
		{
			small+=1;
			total-=5;
		}
		double st = small*0.60;
		cout<<"\t"<<small<<"Small - $"<<st<<endl;
		cout<<"\n\t The Total cost is : $"<<tt+st+mt+bigt;
		}
};
int main(){
Coffee cf;
int bag;
cout<<"\nEnter total no.of coffee bags :"<<endl;
cin>>bag;
cf.div(bag);
return 0;
}
```

Output:



```
Enter total no.of coffee bags :
56

The cost of order : $308
Total boxes used:
        2Large - $3.6
        2Medium - $2
        2Small - $1.2

         The Total cost is : $314.8
--------------------------------
Process exited after 7.285 seconds with return value 0
Press any key to continue . . .
```

LAB TASK No. 5

Task 1: A company called Restaurant Products, Inc. wants to produce small machines for use by waiters in a restaurant. Each machine is initialized with the name of the restaurant and the local sales tax rate. When a customer has finished eating, the waiter enters the cost of the meal and has the machine output a bill in the following form:

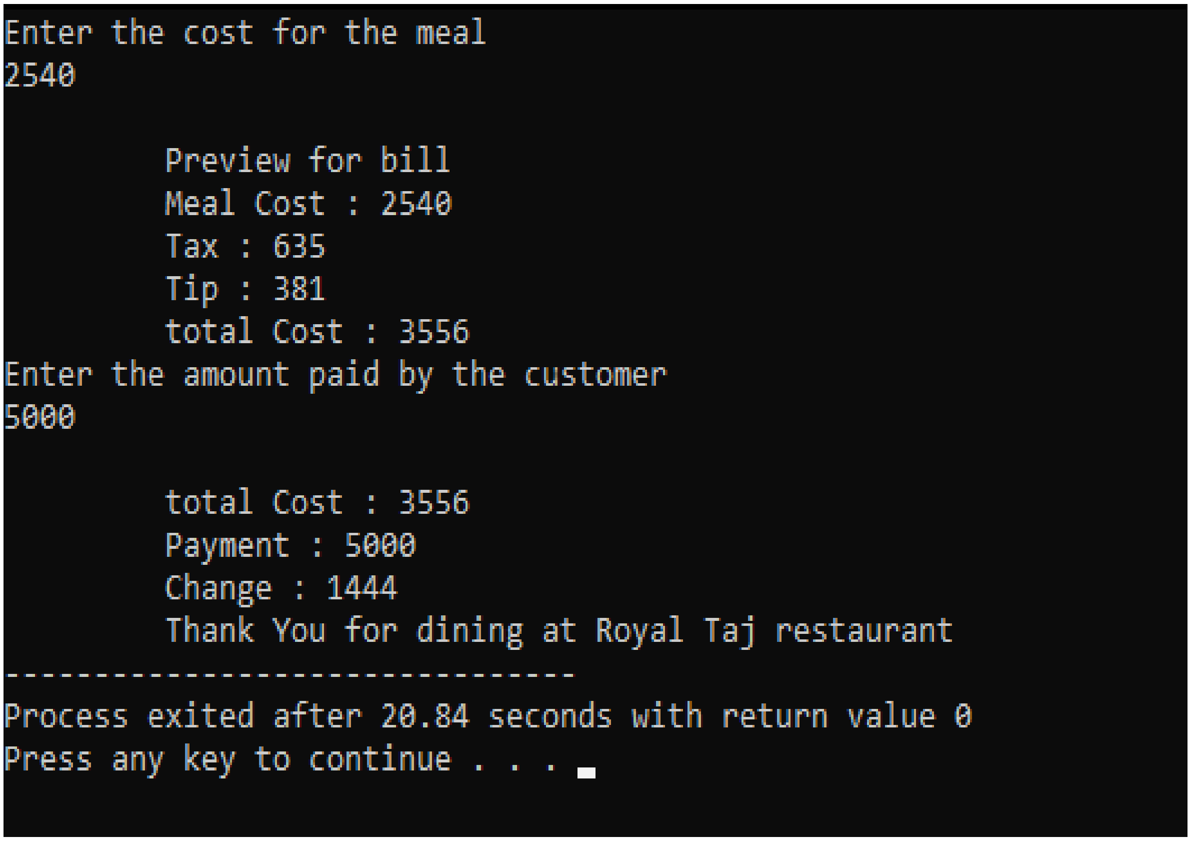
To produce the software for this machine write down the class description and declaration for the `MealBill` class.

Code:

```
#include <iostream>
#include <conio.h>
#include <string.h>
#include <math.h>
#include <cmath>
using namespace std;
class MealBill
{
double tcost;
int tax;
double tip;
public:
void bill(double meal)
{
        tax=meal *0.25;
        tip= meal * 0.15;
        tcost=tip+meal+tax;
        cout<<"\n\t Meal Cost : "<<meal;
        cout<<"\n\t Tax : "<<tax;
        cout<<"\n\t Tip : "<<tip;
        cout<<"\n\t total Cost : "<<tcost;
}
void reciept(double p)
{
        cout<<"\n\t total Cost : "<<tcost;
        cout<<"\n\t Payment : "<<p;
        cout<<"\n\t Change : "<<p-tcost;
        cout<<"\n\t Thank You for dining at Royal Taj restaurant";
}
};
int main()
```

```
{
MealBill M;
double m;
double r;
cout<<"Enter the cost for the meal"<<endl;
cin>>m;
        cout<<"\n\t Preview for bill";
M.bill(m);
cout<<"\nEnter the amount paid by the customer"<<endl;
cin>>r;
M.reciept(r);
return 0;
}
```

Output:



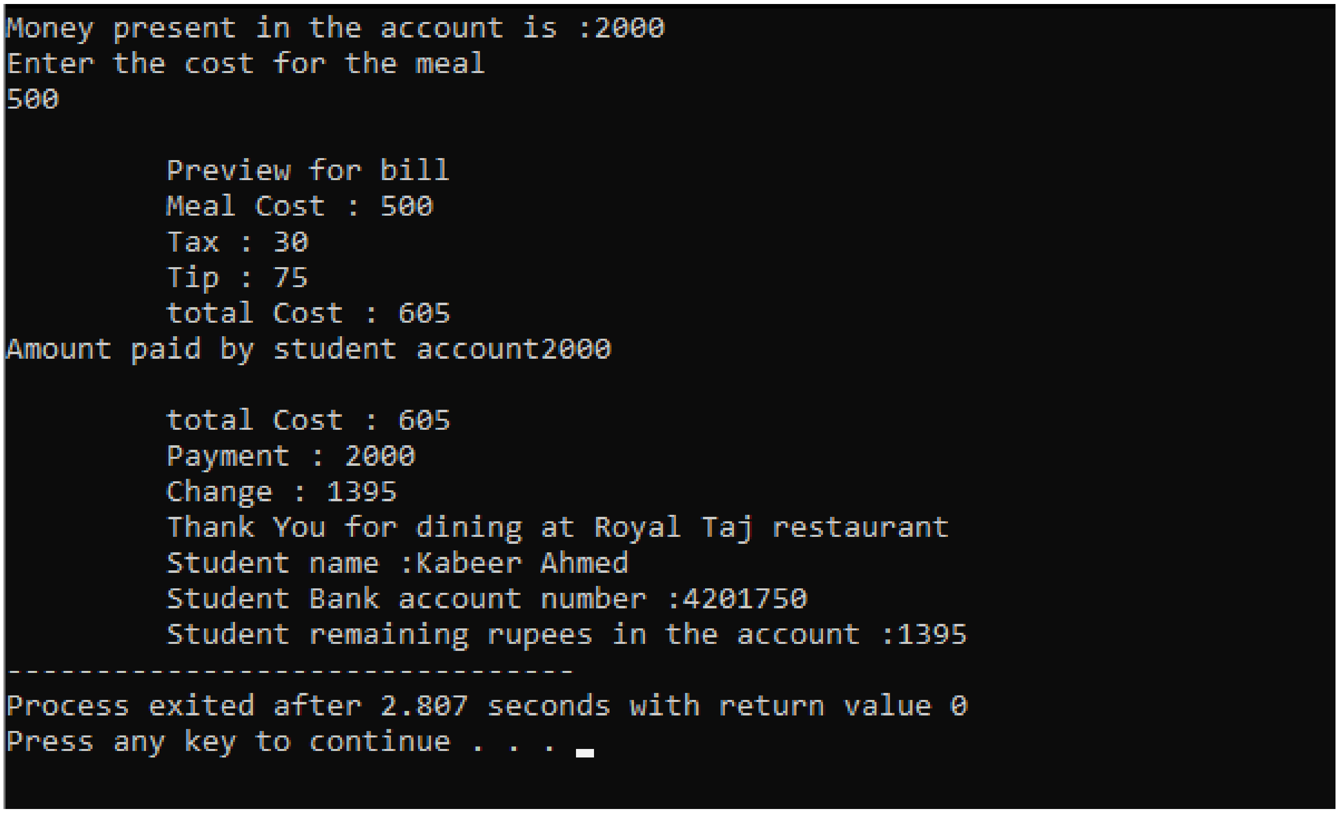
```
Enter the cost for the meal
2540

         Preview for bill
         Meal Cost : 2540
         Tax : 635
         Tip : 381
         total Cost : 3556
Enter the amount paid by the customer
5000

         total Cost : 3556
         Payment : 5000
         Change : 1444
         Thank You for dining at Royal Taj restaurant
--------------------------------
Process exited after 20.84 seconds with return value 0
Press any key to continue . . .
```

Task 2: Write a program for The University Summit Restaurant, which allows a student Ahmed Ali to charge the cost of a meal to his student account. Records are stored in the student account object ahmedAli that has a current balance of Rs.2000.00. The Summit Restaurant implements billing by creating a MealBill object, which is set to charge 6% sales tax. The program first prompts for the cost of the meal and then computes and prints the bill. It then makes a charge to the student's account and prints the receipt. In the end the program prints the current status of Ahmed's account.

Code:

```
#include <iostream>
#include <conio.h>
#include <string.h>
#include <math.h>
#include <cmath>
using namespace std;
class studentBill
{
double tcost;
int tax;
double tip;
public:
string name;
string aname;
studentBill(string n,string a ): name(n),aname(a) {  }
void bill(double meal)
{
        tax=meal *0.06;
        tip= meal * 0.15;
        tcost=tip+meal+tax;
        cout<<"\n\t Meal Cost : "<<meal;
        cout<<"\n\t Tax : "<<tax;
        cout<<"\n\t Tip : "<<tip;
        cout<<"\n\t total Cost : "<<tcost;
}
void reciept(double p)
{
        cout<<"\n\t total Cost : "<<tcost;
        cout<<"\n\t Payment : "<<p;
        cout<<"\n\t Change : "<<p-tcost;
        cout<<"\n\t Thank You for dining at Royal Taj restaurant";
        cout<<"\n\t Student name :"<<name;
        cout<<"\n\t Student Bank account number :"<<aname;
        cout<<"\n\t Student remaining rupees in the account :"<<p-tcost;
}
};
int main()
{
studentBill st("Kabeer Ahmed","4201750");
double a = 2000;
cout<<"Money present in the account is :"<<a<<endl;
double m;
cout<<"Enter the cost for the meal"<<endl;
cin>>m;
cout<<"\n\t Preview for bill";
st.bill(m);
cout<<"\nAmount paid by student account"<<a<<endl;
st.reciept(a);
return 0;
}
```

Output:



```
Money present in the account is :2000
Enter the cost for the meal
500

         Preview for bill
         Meal Cost : 500
         Tax : 30
         Tip : 75
         total Cost : 605
Amount paid by student account2000

         total Cost : 605
         Payment : 2000
         Change : 1395
         Thank You for dining at Royal Taj restaurant
         Student name :Kabeer Ahmed
         Student Bank account number :4201750
         Student remaining rupees in the account :1395
--------------------------------
Process exited after 2.807 seconds with return value 0
Press any key to continue . . .
```

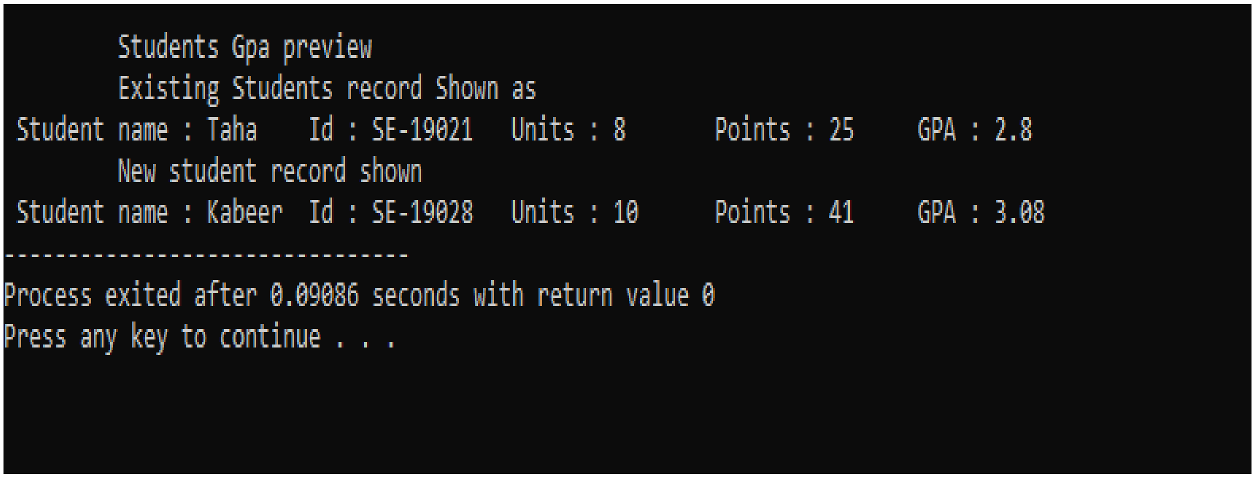
Task 3: A student’s grade record is maintained by the registrar in the Academic Dept. The record includes the studentID along with the total number of credits attempted and the total grade points earned by the student. The grades points are determined by the scale A(4), B(3), C(2), D(1), and F(0).

Code:

```
#include <iostream>
#include <conio.h>
#include <string.h>
#include <math.h>
#include <cmath>
using namespace std;
class gpa
{
string name;
string id;
double GPA;
char gp;
int credit ;
int points;
public :
gpa()
{
        gp=NULL;
        credit=0;
        points=0;
}
gpa(string n , string i, double G , int p , int c) : name(n), id(i) , GPA(G), credit(c), points(p) { }
char AP(char g)
{
  switch(g)
        {
                case 'A':
                        return 4*4;
                        break;
                case 'B':
                        return 4*3;
                        break;
                case 'C':
                        return 4*2;
                        break;
                case 'F':
                        return 4*0;
                        break;
        }
}
int oop(char g)
{
        switch(g)
        {
                case 'A':
                        return 4*4;
                        break;
                case 'B':
                        return 4*3;
                        break;
                case 'C':
                        return 4*2;
                        break;
                case 'F':
                        return 4*0;
                        break;
        }
}
int SE(char g)
{
        switch(g)
        {
                case 'A':
                        return 3*4;
                        break;
                case 'B':
                        return 3*3;
                        break;
                case 'C':
                        return 3*2;
                        break;
                case 'F':
                        return 3*0;
                        break;
        }
}
int ARW(char g)
{
        switch(g)
        {
                case 'A':
                        return 2*4;
                        break;
                case 'B':
                        return 2*3;
                        break;
                case 'C':
                        return 2*2;
                        break;
                case 'F':
                        return 2*0;
                        break;
        }
```

```
}
void show()
{
        cout<<" \n Student name : "<<name;
        cout<<"\tId : "<<id;
        cout<<"\tUnits : "<<credit;
        cout<<"\tPoints : "<<points;
        cout<<"\tGPA : "<<GPA;
}
};
int main()
{
        cout<<"\n\t Students Gpa preview";
        cout<<"\n\t Existing Students record Shown as";
        gpa gg("Taha","SE-19021",2.8,25,8 );
        gg.show();
        gpa gg1;
        int oo=gg.oop('A');
        int ap=gg.AP('C');
        int se=gg.SE('B');
        int arw=gg.ARW('A');
        cout<<"\n\t New student record shown";
                        cout<<" \n Student name : Kabeer";
                        cout<<"\tId : SE-19028";
                        cout<<"\tUnits : "<<(oo+ap+se+arw)/4;
                        cout<<"\tPoints : "<<oo+ap+se+arw;
                        cout<<"\tGPA : "<<3.08;

        return 0;
}
```

Output:



```
        Students Gpa preview
        Existing Students record Shown as
 Student name : Taha    Id : SE-19021   Units : 8       Points : 25     GPA : 2.8
        New student record shown
 Student name : Kabeer  Id : SE-19028   Units : 10      Points : 41     GPA : 3.08
--------------------------------
Process exited after 0.09086 seconds with return value 0
Press any key to continue . . .
```

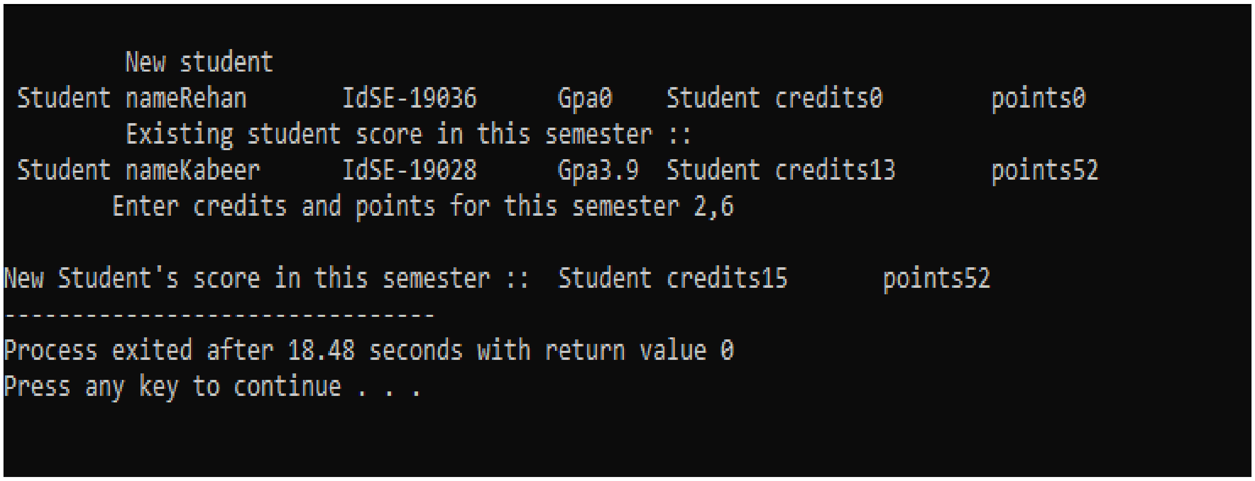
Task 4: Write a program that creates a student grade record for a new student. To indicate that student is new print the students initial grade point average. The program then prompts for the grade points and credits that the student has earned in the first semester and updates his/her grade record. The student’s record at the end of the semester is then output to the screen.

Code:

```
#include <iostream>
#include <conio.h>
#include <string.h>
#include <math.h>
#include <cmath>
using namespace std;
class unistd
{
string name;
string id;
double GPA;
int credit ;
int points;
public:
unistd(string n , string i)
{
        cout<<"\n\t New student ";
        cout<<"\n Student name"<<n;
        cout<<"\t Id"<<i;
        cout<<"\t Gpa"<<0.0;
        cout<<"\t Student credits"<<0;
        cout<<"\t points"<<0;
}
unistd(string n , string i, double G , int p , int c)
{
        cout<<"\n\t Existing student score in this semester ::";
        cout<<"\n Student name"<<n;
        cout<<"\t Id"<<i;
        cout<<"\t Gpa"<<G;
        points=p;
        credit=c;
        cout<<"\t Student credits"<<c;
        cout<<"\t points"<<p;
}
void updated(int p , int c)
{
        cout<<"\nNew Student's score in this semester ::";
        cout<<"\t Student credits"<<c+credit;
        cout<<"\t points"<<p+points;
}
};
```

```
int main(){
unistd s1("Rehan","SE-19036");
unistd s2("Kabeer","SE-19028",3.9,52,13);
int n1,n2;
cout<<"\n\tEnter credits and points for this semester ";
cin>>n1>>n2;
s2.updated(n2,n1);
return 0;
}
```

Output:



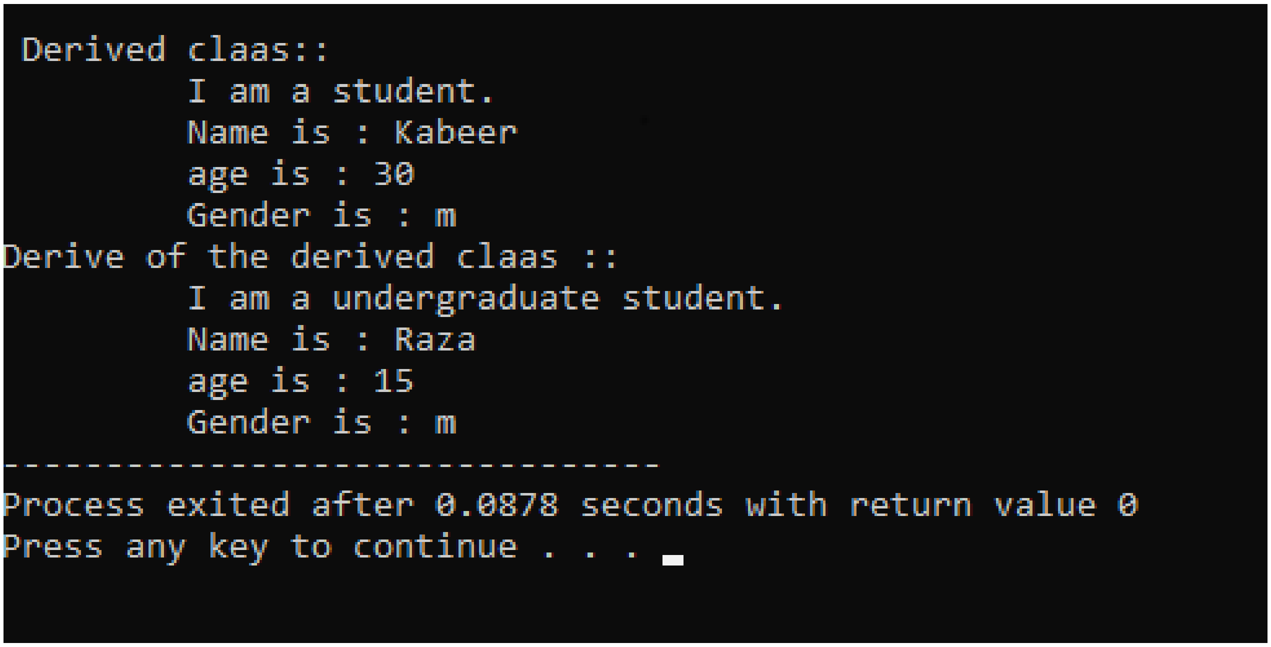
```

         New student
 Student nameRehan       IdSE-19036      Gpa0    Student credits0        points0
         Existing student score in this semester ::
 Student nameKabeer      IdSE-19028      Gpa3.9  Student credits13       points52
        Enter credits and points for this semester 2,6

New Student's score in this semester ::  Student credits15       points52
--------------------------------
Process exited after 18.48 seconds with return value 0
Press any key to continue . . .
```

Task 5: Create a Base Class Person having attributes(name,age,gender) with behavior of showdata(displaying all the members) and override same method in derived classes.Derive student Class and from student class derive graduate student with appropriate data members and behavior.

Code:

```
#include <iostream>
#include <conio.h>
#include <string.h>
#include <math.h>
#include <cmath>
using namespace std;
class person
{
protected:
   string name;
   int age ;
   char gender ;
public:
void show()
{
        cout<<"\n\t Base Class";
        cout<<"\n\t Name is : "<<name;
        cout<<"\n\t age is : "<<age;
        cout<<"\n\t gender isis : "<<gender;
}
};
class student : public person
{
public :
virtual void show(string n , int a , char g)
{
        cout<<"\n Derived claas:: ";
        cout<<"\n\t I am a student.";
        cout<<"\n\t Name is : "<<n;
        cout<<"\n\t age is : "<<a;
        cout<<"\n\t Gender is : "<<g;
}
};
class grastudent : public student
{
public :
virtual void show(string n , int a , char g)
{
        cout<<"\nDerive of the derived claas :: ";
        cout<<"\n\t I am a undergraduate student. ";
        cout<<"\n\t Name is : "<<n;
        cout<<"\n\t age is : "<<a;
        cout<<"\n\t Gender is : "<<g;
}
};
int main()
{
student ss;
ss.show("Kabeer",30,'m');
grastudent gs;
gs.show("Raza",15,'m');
return 0;
}
```

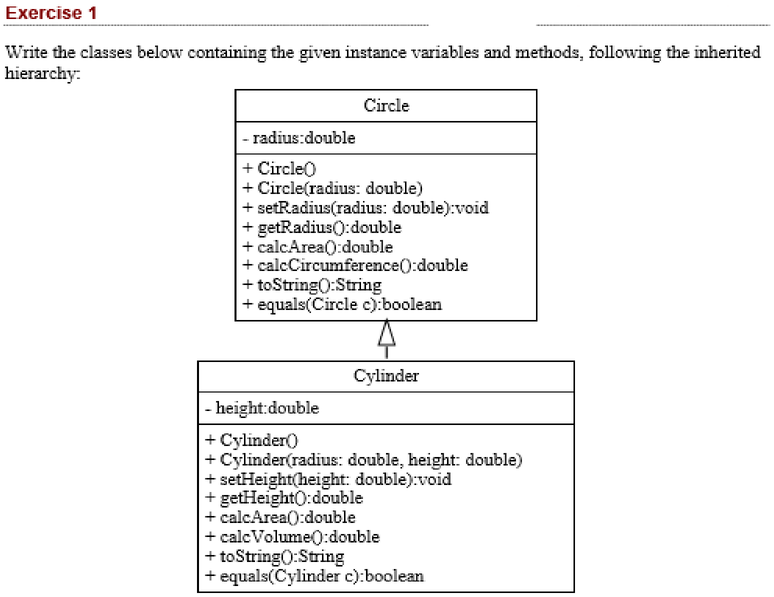
Output:



```
 Derived claas::
         I am a student.
         Name is : Kabeer
         age is : 30
         Gender is : m
Derive of the derived claas ::
         I am a undergraduate student.
         Name is : Raza
         age is : 15
         Gender is : m
--------------------------------
Process exited after 0.0878 seconds with return value 0
Press any key to continue . . .
```

LAB TASK No. 6

Task 1:



Exercise 1

Write the classes below containing the given instance variables and methods, following the inherited hierarchy:

Circle

- radius:double

+ Circle()
+ Circle(radius: double)
+ setRadius(radius: double):void
+ getRadius():double
+ calcArea():double
+ calcCircumference():double
+ toString():String
+ equals(Circle c):boolean

Cylinder

- height:double

+ Cylinder()
+ Cylinder(radius: double, height: double)
+ setHeight(height: double):void
+ getHeight():double
+ calcArea():double
+ calcVolume():double
+ toString():String
+ equals(Cylinder c):boolean

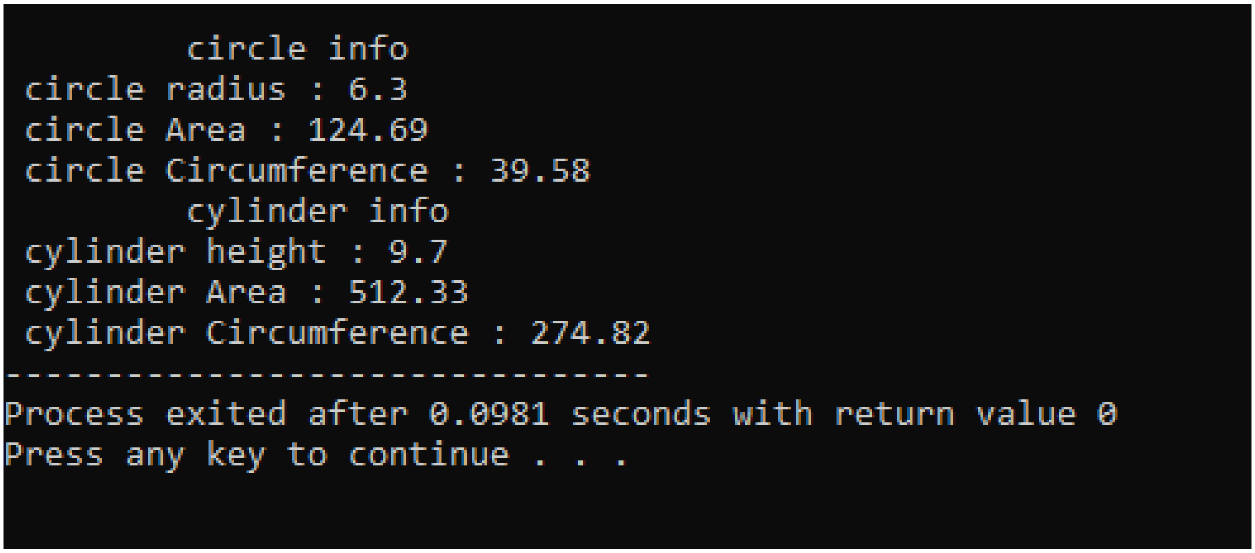
Code:

```
#include <iostream>
#include <conio.h>
#include <string.h>
#include <math.h>
#include <cmath>
using namespace std;
class circle
{
protected:
double radius;
public:
circle(double r): radius(r) {  }
void setradius(double r)
{
        radius=r;
}
double getradius()
{
        return radius ;
}
double Area()
```

```
{
	return 3.14 * pow(radius,2);
}
double circumference()
{
	return 2 * 3.14 * radius;
}
void toSring(double a,double c)
{
	cout<<"\n\t circle info ";
	cout<<"\n circle radius :"<<radius;
	cout<<"\n circle Area :"<<a;
	cout<<"\n circle Circumference :"<<c;
}
};
class cylinder : public circle
{
protected:
double height;
public:
cylinder(double r,double h): radius(r), height (h) {  }
void setheight(double h)
{
	height=h;
}
double getheight()
{
	return height ;
}
virtual double Area()
{
	return ( 2 * 3.14 * radius * height) * (2 * 3.14 * pow(radius,2));
}
virtual	double circumference()
{
	return 3.14 * pow(radius,2) * height;
}
virtual void toSring(double a,double c)
{
	cout<<"\n\t cylinder info ";
	cout<<"\n cylinder height :"<<height;
	cout<<"\n cylinder Area :"<<a;
	cout<<"\n cylinder Circumference :"<<c;
}
};
int main()
{
```

```
circle c1(6.3);
double a= c1.Area();
double c= c1.circumference();
c1.toSring(a,c);
cylinder cc1(5.4,9.7);
double aa= cc1.Area();
double cc= cc1.circumference();
cc1.toSring(aa,cc)
return 0;
}
```

Output:



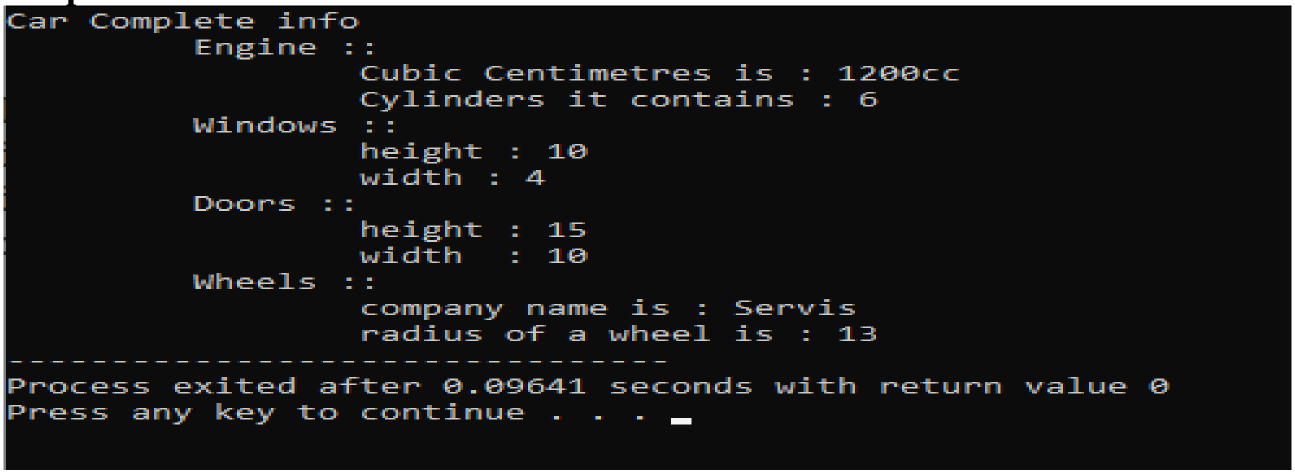
```
         circle info
 circle radius : 6.3
 circle Area : 124.69
 circle Circumference : 39.58
         cylinder info
 cylinder height : 9.7
 cylinder Area : 512.33
 cylinder Circumference : 274.82
--------------------------------
Process exited after 0.0981 seconds with return value 0
Press any key to continue . . .
```

Task 2: Apply the Concept of Composition by Creating part classes (doors,windows,Engine,Wheels) and then Create a whole class Car that is composed of all these defined parts.

Code:

```
#include <iostream>
#include <conio.h>
#include <string.h>
#include <math.h>
#include <cmath>
using namespace std;
class windows
{
int height ;
int width ;
public:
windows(){                    }
windows(int h, int b): height(h), width(b) {          }
void showwindows()
{
	cout<<"\n\t Windows ::";
	cout<<"\n\t\t height : "<<height;
	cout<<"\n\t\t width : "<<width;
}
};
class engine
{
int cc ;
int cylinder ;
public:
engine(){                     }
engine(int c,int c1): cc(c) , cylinder(c1) {            }
void showengine()
{
	cout<<"\n\t Engine ::";
	cout<<"\n\t\t Cubic Centimetres is : "<<cc<<"cc";
	cout<<"\n\t\t Cylinders it contains : "<<cylinder;
}
};
class doors
{
int height ;
int width ;
public:
doors(){                      }
doors(int h, int b): height(h), width(b) {  }
void showdoors()
{
	cout<<"\n\t Doors ::";
	cout<<"\n\t\t height : "<<height;
	cout<<"\n\t\t width  : "<<width;
}
};
class wheels
{
double radius ;  string cmname ;
public:
wheels(){                    }
wheels(double r, string name): radius(r), cmname(name) {          }
void showwheels()
{
	cout<<"\n\t Wheels ::";
	cout<<"\n\t\t company name is : "<<cmname;
	cout<<"\n\t\t radius of a wheel is : "<<radius;
}
};
class Car{
engine eng; windows win; doors dr; wheels whe;
public:
Car(engine e, windows w, doors d, wheels w1): eng(e),win(w),dr(d), whe(w1) {   }
void showinfo()
{
	cout<<"Car Complete info";
	eng.showengine();
	win.showwindows();
	dr.showdoors();
	whe.showwheels();
}
};
int main(){
	engine e(1200,6);
	windows w(10,4);
	doors d(15,10);
	wheels wh(13,"Servis");
	Car ccc(e,w,d,wh);
	ccc.showinfo();
	return 0;}
```

Output:



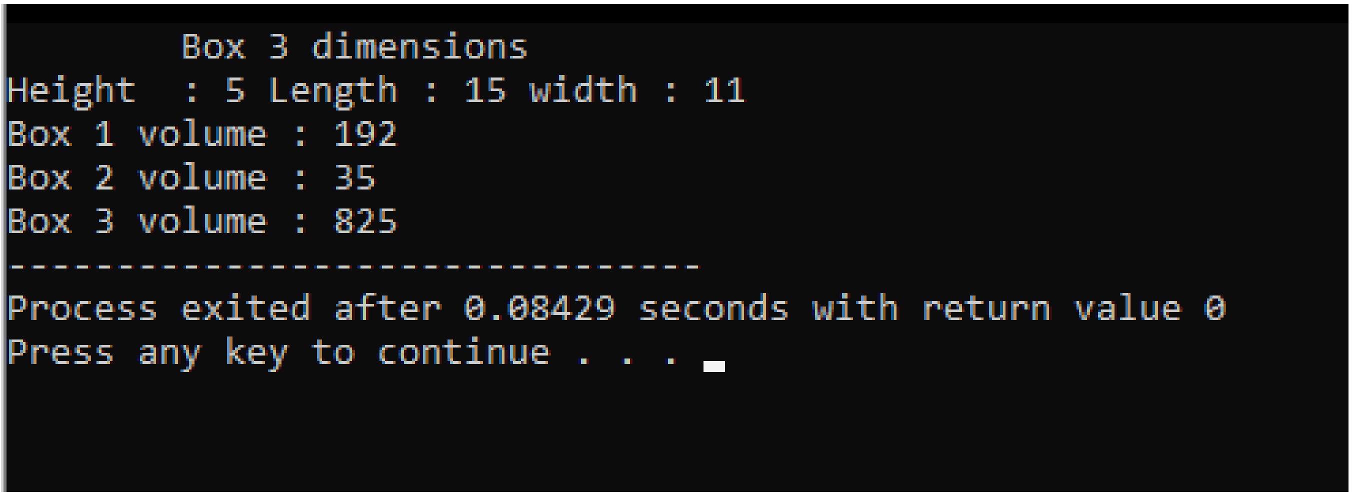
```
Car Complete info
         Engine ::
                 Cubic Centimetres is : 1200cc
                 Cylinders it contains : 6
         Windows ::
                 height : 10
                 width : 4
         Doors ::
                 height : 15
                 width  : 10
         Wheels ::
                 company name is : Servis
                 radius of a wheel is : 13
--------------------------------
Process exited after 0.09641 seconds with return value 0
Press any key to continue . . . _
```

Task 3: Create Box class containing data members (width ,height,Length) and Member functions showdata and Volume.. Apple the concept of Operator Overloading and overload operators (+,-,* ,/) for addition subtraction multiplication and division of Objects of Box class.

Code:

```
#include <iostream>
#include <conio.h>
#include <string.h>
#include <math.h>
#include <cmath>
using namespace std;
class Box
{
int height;
int width;
int length;
public:
Box(){};
Box(int h, int w, int l) : height(h), width(w), length(l){};
int getheight()
{
   return height;
}
void setheight(int h)
{
   height = h;
}
int getwidth()
{
   return width;
}
void setwidth(int w)
{
   width = w;
}
int getlength()
{
   return length;
}
void setlength(int l)
{
   length = l;
}
int volume()
{
   return height * length * width;
}
Box operator*(const Box b)
{
   Box box;
   box.height = this->height * b.height;
   box.width = this->width * b.width;
   box.length = this->length * b.length;
   return box;
}
Box operator+(const Box b)
{
   Box box;
   box.height = this->height + b.height;
   box.width = this->width + b.width;
   box.length = this->length + b.length;
   return box;
}
Box operator/(const Box b)
{
   Box box;
   box.height = this->height / b.height;
   box.width = this->width / b.width;
   box.length = this->length / b.length;
   return box;
}
Box operator-(const Box &b)
{
   Box box;
   box.height = this->height - box.height;
   box.width = this->width - box.width;
   box.length = this->length - box.length;
   return box;
}
};
int main()
{
Box b1(4,6,8);
Box b2(1,5,7);
Box b3;
b3=b1+b2;
cout<<"\tBox 3 dimensions"<<endl;
cout<<"Height  : "<<b3.getheight()<<" Length : "<<b3.getlength()<<" width : "<< b3.getwidth();
cout<<endl<<"Box 1 volume : "<<b1.volume();
cout<<endl<<"Box 2 volume : "<<b2.volume();
cout<<endl<<"Box 3 volume : "<<b3.volume();
return 0;
}
```

Output:



```
        Box 3 dimensions
Height  : 5 Length : 15 width : 11
Box 1 volume : 192
Box 2 volume : 35
Box 3 volume : 825
--------------------------------
Process exited after 0.08429 seconds with return value 0
Press any key to continue . . .
```

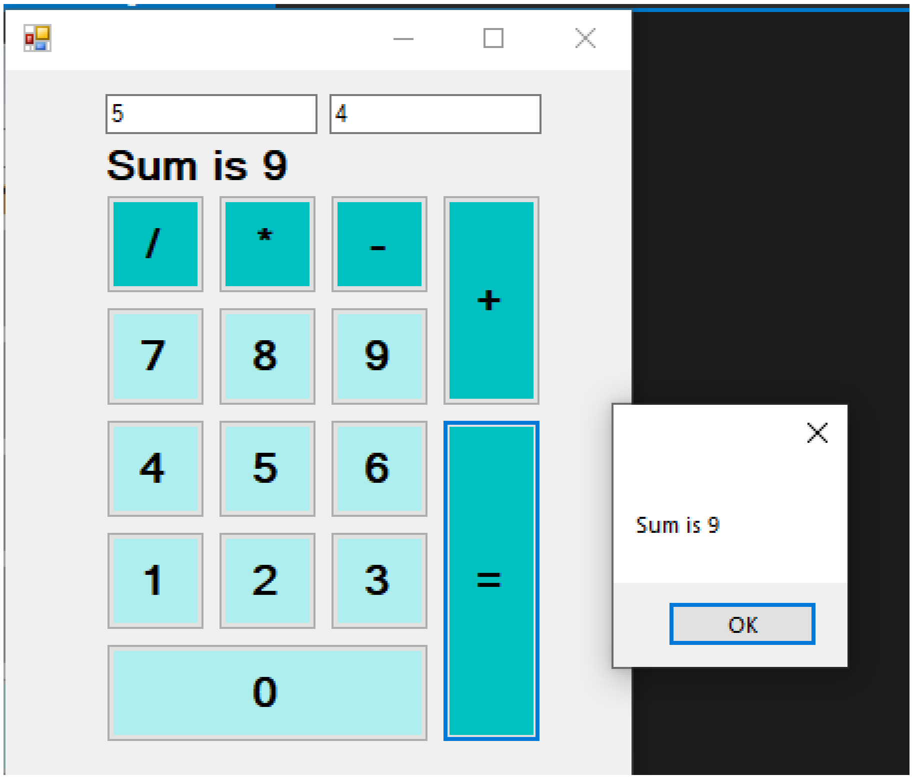
LAB TASK No. 7

Task 1: Create a Calculator class that offers four methods. Add, subtract, multiply and Divide. Consisting of two private members of type double to take input from the user. Create object of a class and start using the calculator class.

Code:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace WindowsFormsApp4
{
    public partial class Form1 : Form
    {
        class calculator
        {
            private int num1;
            private int num2;
            public calculator() { }
            public int add(int n, int n2)
            {
                return n + n2;
            }
            public int subs(int n, int n2)
            {
                return n - n2;
            }
            public int prod(int n, int n2)
            {
                return n * n2;
            }
            public int div(int n, int n2)
            {
                return n / n2;
            }
        }
        public Form1()
        {
            InitializeComponent();
        }
        private void Form1_Load(object sender, EventArgs e)
        {
            label1.Text = "Output";
        }
        private void button4_Click(object sender, EventArgs e)
        {
            int n1;
            int n2;
            calculator c = new calculator();
            bool num1 = int.TryParse(textBox1.Text, out n1);
            bool num2 = int.TryParse(textBox2.Text, out n2);
            int result = c.add(n1, n2);
            label1.Text = "Sum is " + result;
        }
        private void button1_Click(object sender, EventArgs e)
        {
            int n1;
            int n2;
            calculator c = new calculator();
            bool num1 = int.TryParse(textBox1.Text, out n1);
            bool num2 = int.TryParse(textBox2.Text, out n2);
            int result = c.subs(n1, n2);
            label1.Text ="Subtraction  is " + result;

        }
        private void button3_Click(object sender, EventArgs e)
        {
            int n1;
            int n2;
            calculator c = new calculator();
            bool num1 = int.TryParse(textBox1.Text, out n1);
            bool num2 = int.TryParse(textBox2.Text, out n2);
            int result = c.prod(n1, n2);
            label1.Text = "Product  is " + result;
        }

        private void button2_Click(object sender, EventArgs e)
        {
            int n1;
            int n2;
            calculator c = new calculator();
            bool num1 = int.TryParse(textBox1.Text, out n1);
            bool num2 = int.TryParse(textBox2.Text, out n2);
            int result = c.div(n1, n2);
            label1.Text = "Division  is " + result;
        }
        private void button5_Click(object sender, EventArgs e)
        {
            MessageBox.Show(label1.Text);
        }
    }
}
```

Output:



5

4

Sum is 9

Sum is 9

LAB TASK No. 8

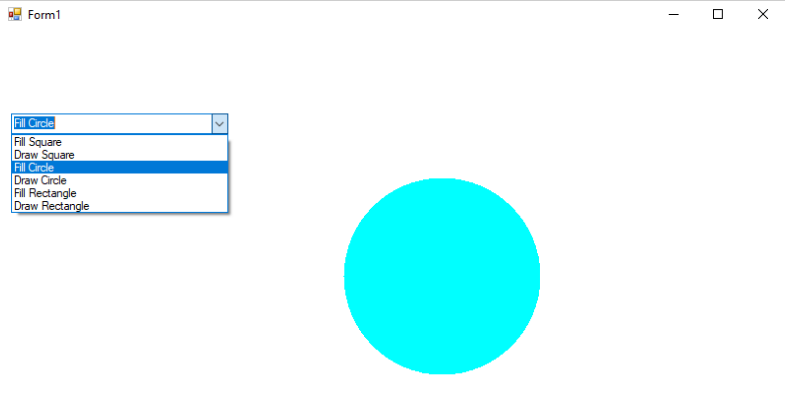
Task 1:

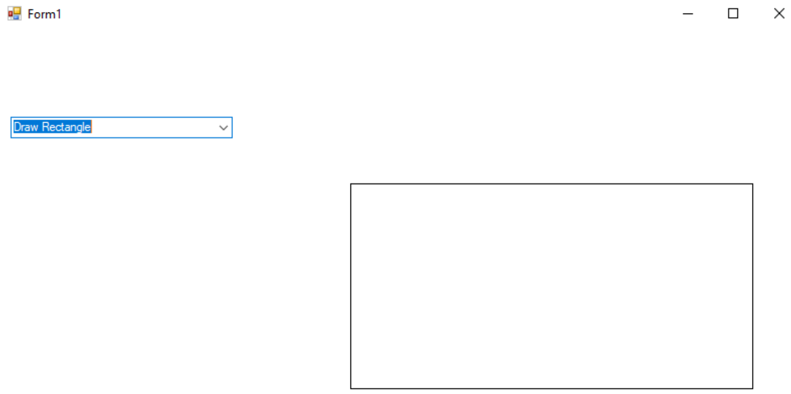
Create a simple C# that will place one combo box filled with the names of different shapes by using item property of combo box and application should be able to draw the selected shape of combo box control by creating graphic ,pen and solid brush objects.

Code:

```
using System;
using System.Drawing;
using System.Windows.Forms;
namespace WindowsFormsApp3
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
        private void comboBox1_SelectedIndexChanged(object sender, EventArgs e)
        {
            Graphics g = base.CreateGraphics();
            Pen p = new Pen(Color.Black);
            SolidBrush sb = new SolidBrush(Color.Aqua);
            g.Clear(Color.White);
            switch(comboBox1.SelectedIndex)
            {
                case 0:
                    g.FillRectangle(sb, 350, 150, 200, 200);
                    break;
                case 1:
                    g.DrawRectangle(p, 350, 150, 200, 200);
                    break;
                case 2:
                    g.FillEllipse(sb, 350, 150, 200, 200);
                    break;
                case 3:
                    g.DrawEllipse(p, 350, 150, 200, 200);
                    break;
                case 4:
                    g.FillRectangle(sb, 350, 150, 400,200);
                    break;
                case 5:
                    g.DrawRectangle(p, 350, 150, 400, 200);
                    break;
            }
        }
    }
}
```

Design:


Form1
Fill Circle
Fill Square
Draw Square
Fill Circle
Draw Circle
Fill Rectangle
Draw Rectangle


Form1
Draw Rectangle

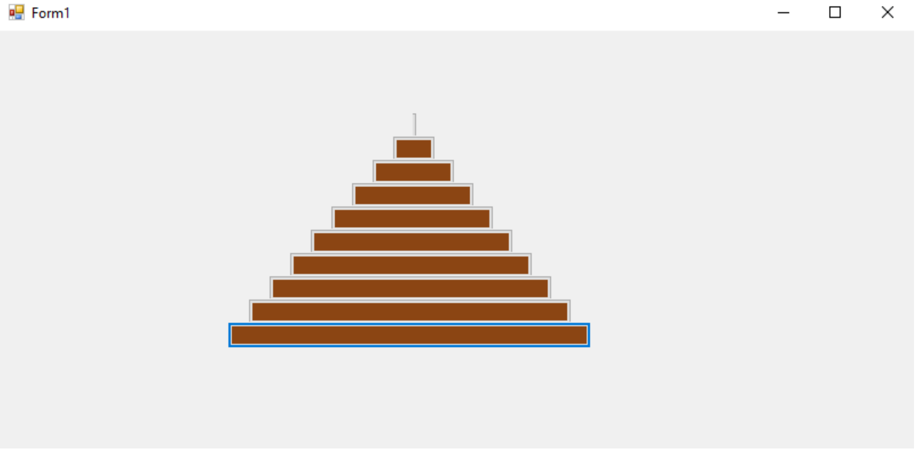
Task 2:

Create an array of Buttons that would display buttons dynamically at run time.

Code:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
namespace WindowsFormsApp3
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void Form1_Load(object sender, EventArgs e)
        {
            Button[] b = new Button[10];
            for (int i = 0; i <= b.Length - 1; i++)
            {
                b[i] = new Button();
                b[i].Text = "";
                b[i].Size = new Size(318 - 35 * i, 23);
                b[i].Location = new Point(200 + 18 * i, 250 - 20 * i);
                b[i].BackColor = System.Drawing.Color.SaddleBrown;
                this.Controls.Add(b[i]);
            }
        }
    }
}
```

Design:



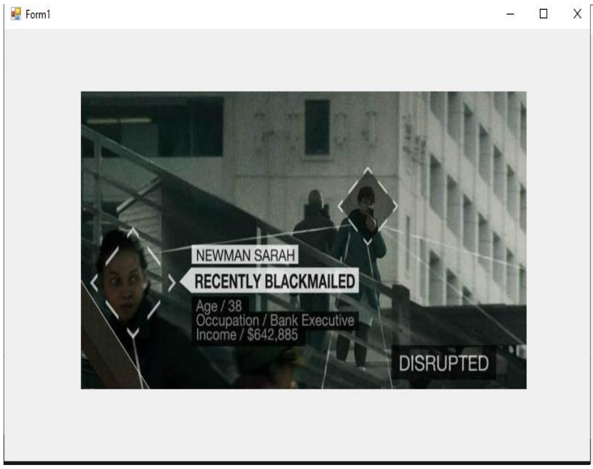
Task 3:

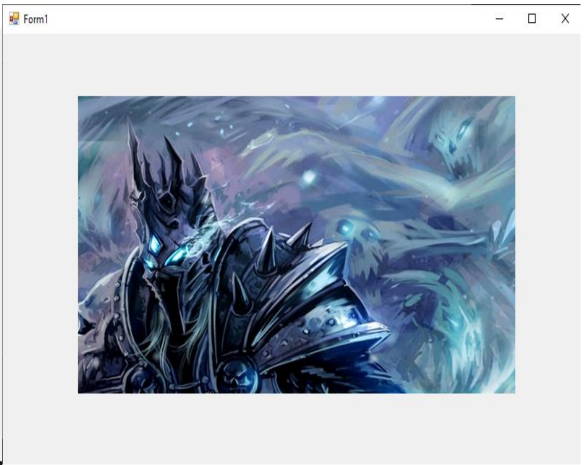
Create a slideshow of pictures by using picture box control and timer control. Make a folder on any drive containing all of your pictures, you want to include in a slideshow. Rename all the pictures starting with 1.jpg, 2.jpg… n.jpg.

Code:

```
using System;
using System.Windows.Forms;
namespace WindowsFormsApp3
{
    public partial class Form1 : Form
    {

        public Form1()
        {
            InitializeComponent();
        }
        private void Form1_Load(object sender, EventArgs e)
        {

        }
        int count = 0;
        private void timer1_Tick(object sender, EventArgs e)
        {
            if(count==10)
            {
                count = 1;
            }
            pictureBox1.ImageLocation = string.Format(@"g:/slideshow/{0}.jpg", count);
            count++;
        }
    }
}
```

Design:





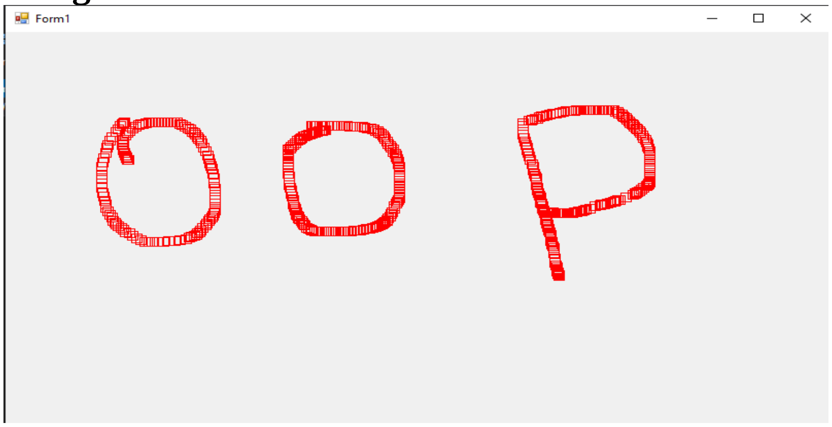
Task 4:

Using Mouse up, down and Mouse move events, create an application that can draw freehand drawing.

Code:

```
using System;
using System.Drawing;
using System.Windows.Forms;
namespace WindowsFormsApp3
{
    public partial class Form1 : Form
    {
        bool isMouseDown = false;
        public Form1()
        {
            InitializeComponent();
        }
        private void Form1_Load(object sender, EventArgs e)
        {
        }
        private void Form1_MouseUp(object sender, MouseEventArgs e)
        {
            isMouseDown = false;
        }
        private void Form1_MouseMove(object sender, MouseEventArgs e)
        {
            if (isMouseDown == true)
            {
                Graphics g = CreateGraphics();
                Pen p = new Pen(Color.Red);
                g.DrawRectangle(p, e.X, e.Y, 10, 10);
            }
        }
        private void Form1_MouseDown(object sender, MouseEventArgs e)
        {
            isMouseDown = true;
        }
    }
}
```

Design:



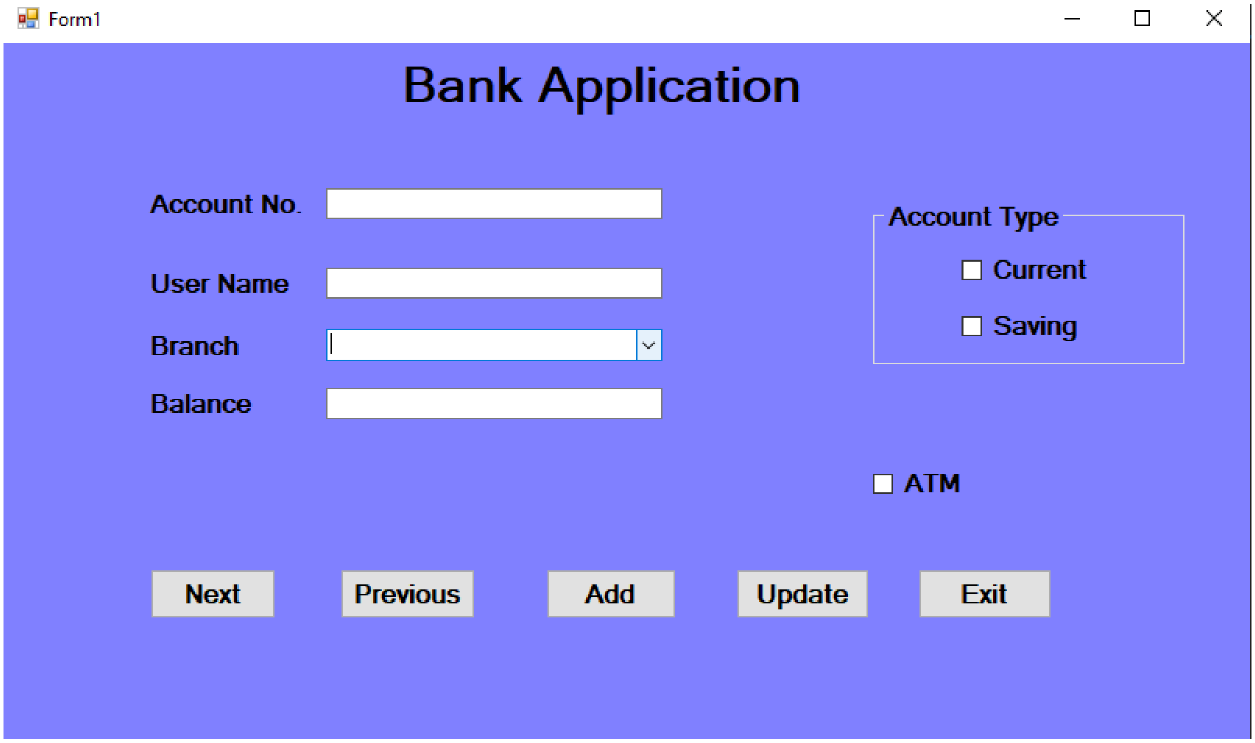
LAB TASK No. 9

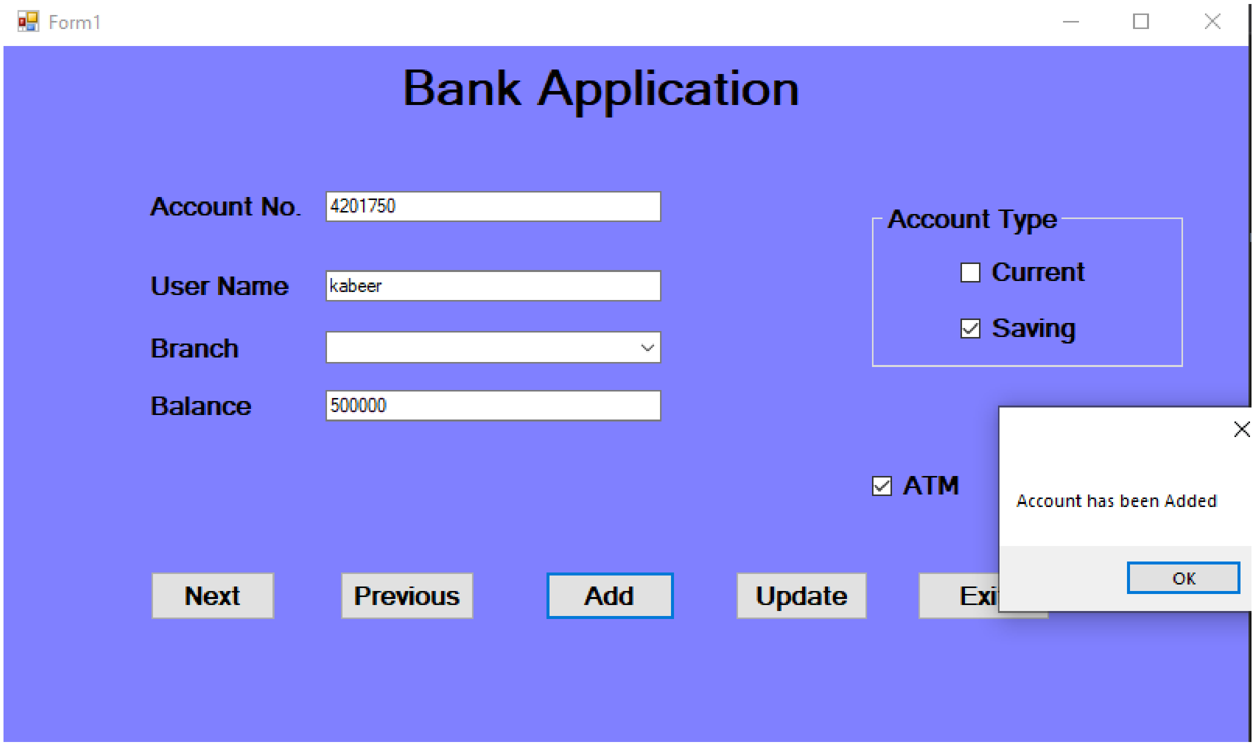
Task 1:

Code:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Data.OleDb;
namespace Accessatabase2
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
        int counter = 0;
        OleDbConnection co = new OleDbConnection(@"Provider=Microsoft.ACE.OLEDB.12.0
Source = E:\bank.accdb");
        OleDbDataAdapter dp = new OleDbDataAdapter("select * from Account",
@"Provider=Microsoft.ACE.OLEDB.12.0;Data Source = E:\bank.accdb")
        DataSet dt = new DataSet("Account");

        private void Form1_Load(object sender, EventArgs e)
        {
            this.accountTableAdapter.Fill(this.dataSet1.Account);
            co.Open();
            dp.Fill(dt);
        }
private void button1_Click(object sender, EventArgs e)
        {
            if (counter < dt.Tables["Account"].Rows.Count - 1)
            {
                counter = counter + 1;
                textBox1.Text = dt.Tables["Account"].Rows[counter]["AccID"].ToString();
                textBox2.Text = dt.Tables["Account"].Rows[counter]["name"].ToString();
                textBox3.Text =
dt.Tables["Account"].Rows[counter]["branchcode"].ToString();
                textBox4.Text = dt.Tables["Account"].Rows[counter]["Balance"].ToString();
            }
            else if (counter <= dt.Tables["Account"].Rows.Count - 1)
            {
                MessageBox.Show("You have already account");
            }

        }
private void button2_Click(object sender, EventArgs e)
        {
            if (counter > 0)
            {
                counter = counter - 1;
                textBox1.Text = 
dt.Tables["Account"].Rows[counter]["AccountNo"].ToString();
                textBox2.Text = dt.Tables["Account"].Rows[counter]["Name"].ToString();
                textBox3.Text = dt.Tables["Account"].Rows[counter]["Balance"].ToString();
                textBox4.Text = dt.Tables["Account"].Rows[counter]["Branch"].ToString();
            }
            else if (counter == 0)
            {
                MessageBox.Show("You have already an account");
            }


        }
private void button4_Click(object sender, EventArgs e)
        {
            OleDbCommand udp = new OleDbCommand("Update Account set Balance='" + 
textBox4.Text + "' where AccID='" + textBox1.Text +"'", co);
            udp.ExecuteNonQuery();
            MessageBox.Show(" Account has been updated");


        }
private void button3_Click(object sender, EventArgs e)
        {
            OleDbCommand ins = new OleDbCommand("Insert into Account(AccID,name, 
branchcode, Balance) values('" + textBox1.Text + "','" + textBox2.Text + "','" + 
textBox3.Text + "','" + textBox4.Text + "')", co);
            ins.ExecuteNonQuery();
            MessageBox.Show(" Account has been Added");

        }

        private void button6_Click(object sender, EventArgs e)
        {
            this.Close();
        }
    }
}
```

Output:





Account No. 4201750

User Name kabeer

Branch

Balance 500000

☑ Saving

☑ ATM

Account has been Added