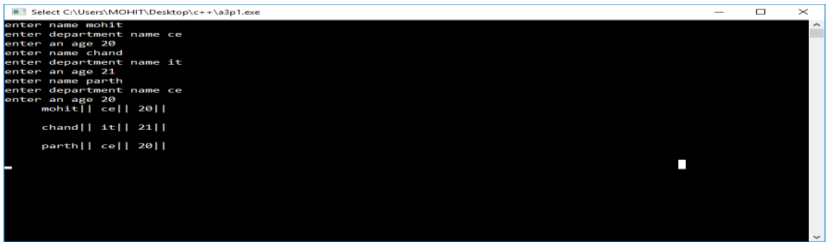
//program 1 */ **Create a class Employee that would describe the following information:**

-Employee Name
-Employee Department
-Employee Age

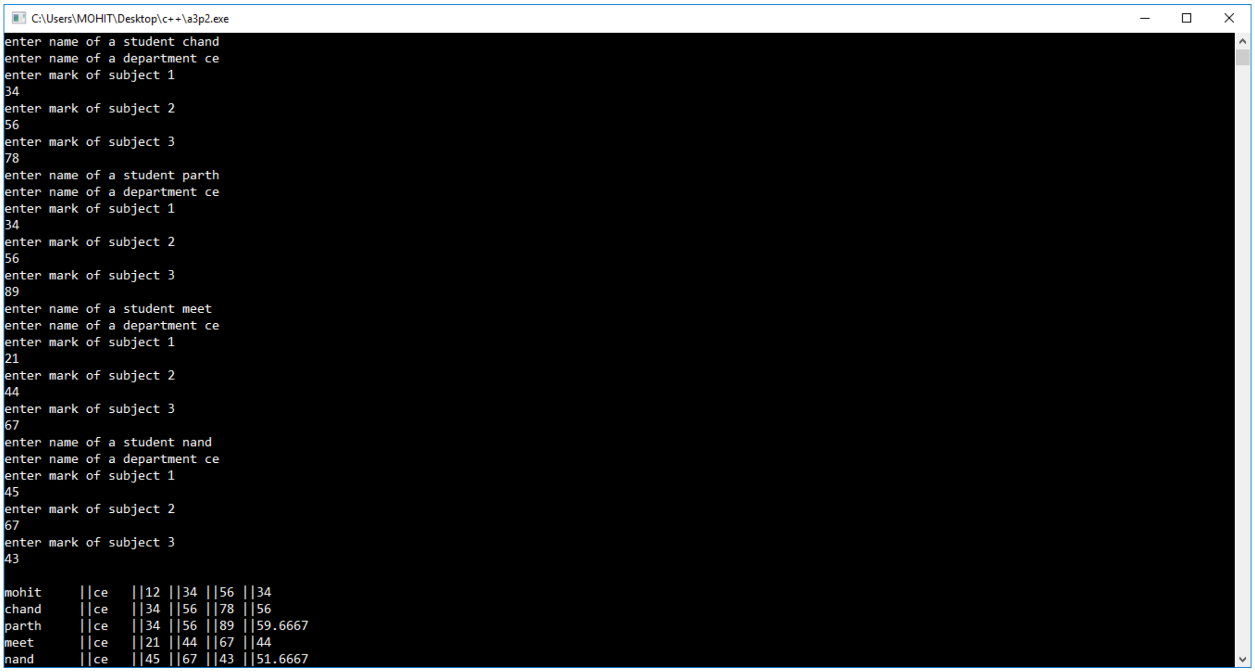
Develop a program using 2 member functions get_record() and put_record() to scan and print the records of employee respectively. Use arrays of object to create a record of 5 employees and print them onto the screen. /*

```
#include<iostream>
#include<stdlib.h>
#include<iomanip>
#include<stdio.h>
#include<conio.h>
using namespace std;
    class employee{
      char name[20];
      char depart[20];
      int age;
          public:
      void get_record()
      {
          fflush(stdin);
        cout << "enter name ";
          cin >> name;
        cout << "enter department name ";
         cin >> depart;
        cout << "enter an age ";
```

```
                cin >> age;

            }

            void put_record()

            {

              cout << setw(10) << name << "|| " << setw(-5) << depart << "|| " << setw(-5) << age << "|| " << endl;

            }

        };

        int main()

            {

                       employee i[5];

                int j;


                    for(j=0;j<3;j++)

                    {

                       i[j].get_record();

                       system("cls");

                    }

                    for(j=0;j<3;j++)

                    {

                       i[j].put_record();

                       cout << endl;

                    } return 0; }
```



```
enter name mohit
enter department name ce
enter an age 20
enter name chand
enter department name it
enter an age 21
enter name parth
enter department name ce
enter an age 20
     mohit|| ce|| 20||

     chand|| it|| 21||

     parth|| ce|| 20||
// program 2/* "Define a  class Student that would describe the
following information:

-Student name

-Department

-3 subject marks

-Percentage

Develop a program using 3 functions getdata(), putdata() and
calculate_avg() for scanning, printing and calculating percentage of 5
students respectively.

Use scope resolution operator" */

#include<iostream>

#include<conio.h>

#include<stdio.h>

#include<iomanip>

#include<stdlib.h>

using namespace std;

    class student{

        char name[20],depart[20];

        float percentage;

        float mark[3];

      public:

        void getdata();

        void putdata();

        void calculate_avg();

    };
    void student::  getdata()  {
        fflush(stdin);
        cout << "enter name of a student ";
          gets(name);
        cout << "enter name of a department ";
          gets(depart);
        for(int i=0;i<3;i++)
        {
          cout << "enter mark of subject " << i+1 << endl;;
            cin >> mark[i];
        }
     }
    void student:: putdata(){
cout<<endl<<setw(10)<<left<<name<<"||"<<setw(5)<<left<<depart<<"||"<<setw(3)<<mark[0]
<<"||"<<setw(3)<<mark[1]<<"||"<<setw(3)<<mark[2]<<"||"<<setw(5)<<percentage;
    }
    void student :: calculate_avg(){
        float total=mark[0]+mark[1]+mark[2];
        percentage=(float)total/3;
    int main()
     {
      student i[5];
          for(int j=0;j<3;j++)
          {
            i[j].getdata();
  i[j].calculate_avg();
            system("cls");
          }
          for(int j=0;j<3;j++)

          {

            i[j].putdata();

          } return 0;  }
```

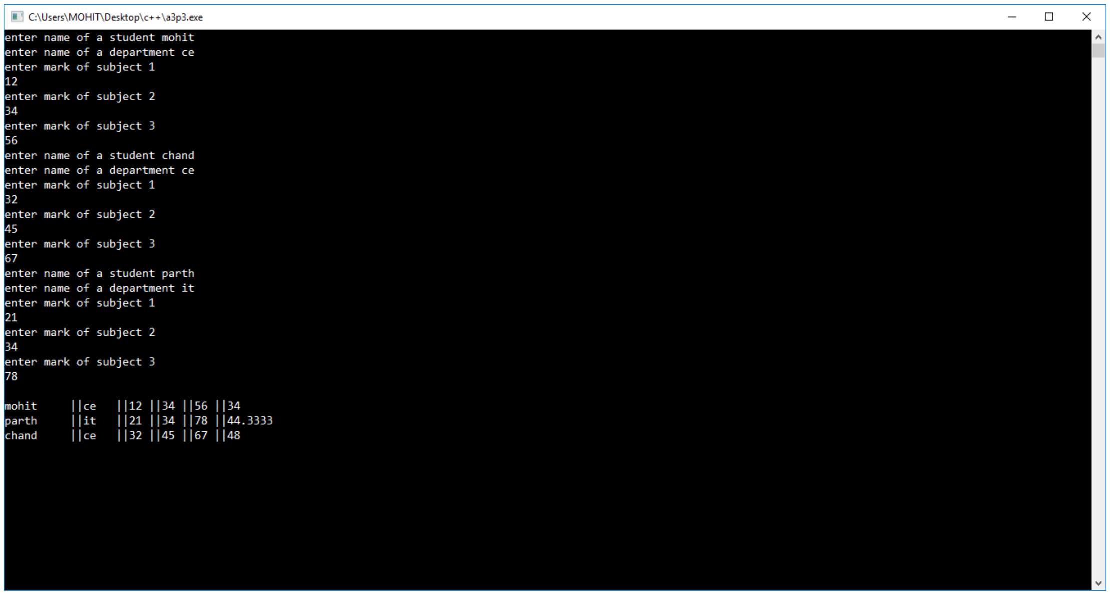


```
enter name of a student chand
enter name of a department ce
enter mark of subject 1
34
enter mark of subject 2
56
enter mark of subject 3
78
enter name of a student parth
enter name of a department ce
enter mark of subject 1
34
enter mark of subject 2
56
enter mark of subject 3
89
enter name of a student meet
enter name of a department ce
enter mark of subject 1
21
enter mark of subject 2
44
enter mark of subject 3
67
enter name of a student nand
enter name of a department ce
enter mark of subject 1
45
enter mark of subject 2
67
enter mark of subject 3
43

mohit     ||ce   ||12 ||34 ||56 ||34
chand     ||ce   ||34 ||56 ||78 ||56
parth     ||ce   ||34 ||56 ||89 ||59.6667
meet      ||ce   ||21 ||44 ||67 ||44
nand      ||ce   ||45 ||67 ||43 ||51.6667
```

// program 3 // Add a function sortdata() to print the students information according to their percentage in ascending order in program 2.

```
#include<iostream>
#include<conio.h>
#include<stdio.h>
#include<iomanip>
#include<stdlib.h>
using namespace std;
    class student{
        char name[20],depart[20];
        float percentage;
        float mark[3];
      public:
        void getdata();
        void putdata();
        void calculate_avg();
        void sort_data();

    };
    student i[5];
   void student::  getdata()  {
        fflush(stdin);
        cout << "enter name of a student ";
          gets(name);
        cout << "enter name of a department ";
          gets(depart);
        for(int i=0;i<3;i++)
        {
          cout << "enter mark of subject " << i+1 << endl;;
            cin >> mark[i];
        }
     }
    void student:: putdata(){

cout<<endl<<setw(10)<<left<<name<<"||"<<setw(5)<<left<<depart<<"||"<<setw(3)<<mark[0]
<<"||"<<setw(3)<<mark[1]<<"||"<<setw(3)<<mark[2]<<"||"<<setw(5)<<percentage;


    }
    void student :: calculate_avg(){
        float total=mark[0]+mark[1]+mark[2];
        percentage=(float)total/3;
    }


    void student::sort_data(){
        int j,m;
              for(m=0;m<3;m++)
          {
            for(j=0;j<=3;j++)
            {
              if(i[m].percentage<i[j].percentage)
```

```
                 {
                     student temp=i[j];
                    i[j]=i[m];
                    i[m]=temp;
                 }
               }
            }
}

    int main()
     {
       //student i[5];
            for(int j=0;j<3;j++)
            {
               i[j].getdata();
               i[j].calculate_avg();
               system("cls");
            }
            for(int j=0;j<3;j++)
            {

                 i[j].sort_data();
            }
            for(int j=0;j<3;j++)
            {
            i[j].putdata();
            }   return 0;   }
```



```
enter name of a student mohit
enter name of a department ce
enter mark of subject 1
12
enter mark of subject 2
34
enter mark of subject 3
56
enter name of a student chand
enter name of a department ce
enter mark of subject 1
32
enter mark of subject 2
45
enter mark of subject 3
67
enter name of a student parth
enter name of a department it
enter mark of subject 1
21
enter mark of subject 2
34
enter mark of subject 3
78

mohit     ||ce    ||12 ||34 ||56 ||34
parth     ||it    ||21 ||34 ||78 ||44.3333
chand     ||ce    ||32 ||45 ||67 ||48
```

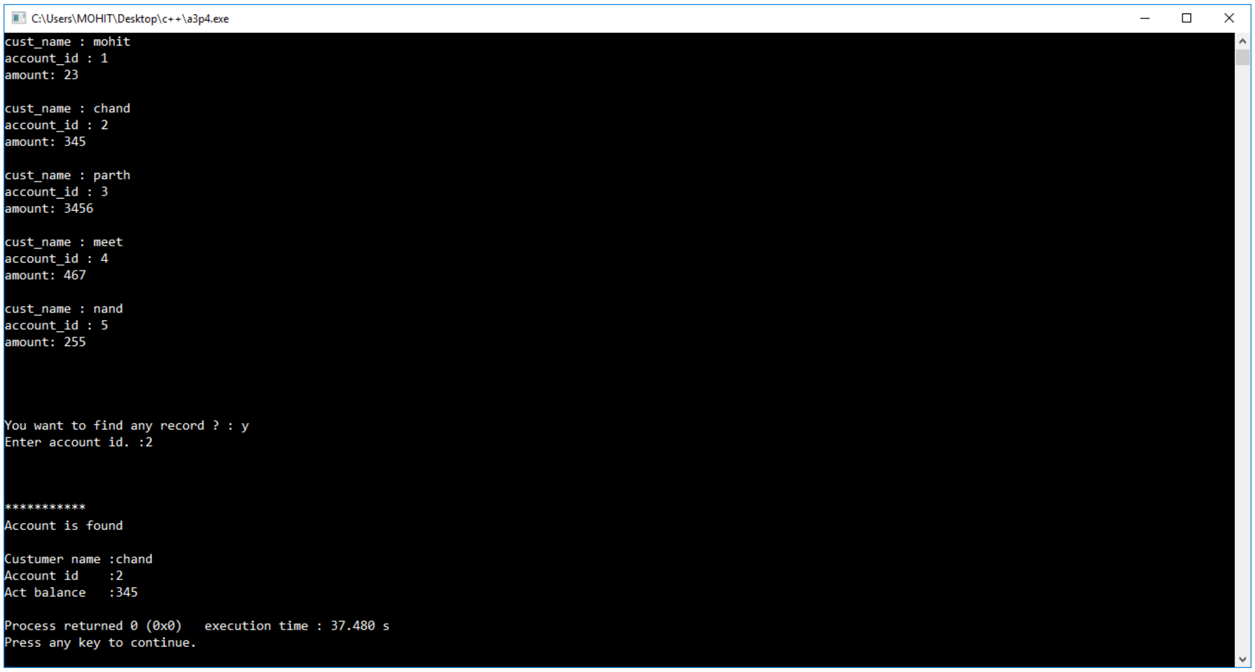
```
// programme 4
/* "Create a class Account. It has three data member account id, name and balance.
Define function to assign value and display value. Define function that search
account number given by the user. If account number exists, print detail of that account. Write a program using array of object. Declare at least 5 account and print details."
*/
#include<iostream>
#include<conio.h>
#include<stdio.h>
#include<stdlib.h>
using namespace std;

class account
{
	int account_id;
	char cust_name[20];
	int balance;

	public:
		void getdata();
		void putdata();
```

```
                    int search(int);
};
void account::getdata()
{
          balance=0;

          cout<<"Enter name :";
          cin>>cust_name;

          cout<<"Enter act_id :";
          cin>>account_id;

          cout<<"Enter balance :";
          cin>>balance;

          cout<<"\n";
}

void account::putdata()
{
          cout<<"cust_name : "<<cust_name<<endl;
          cout<<"account_id : "<<account_id<<endl;
          cout<<"amount: "<<balance<<endl;
          cout<<"\n";
}

int account::search(int data)
```

```
{
        int flag;

        if(data==account_id)
        {
                flag=1;
                cout<<"\n\n\n***********\nAccount is found\n\n";
                cout<<"Custumer name :"<<cust_name<<endl;
                cout<<"Account id    :"<<account_id<<endl;
                cout<<"Act balance   :"<<balance<<endl;
        }

        return flag;

}


account a[5];

int main()
{
        int i,data,flag=0;
        char c;



        for(i=0;i<5;i++)
        {
```

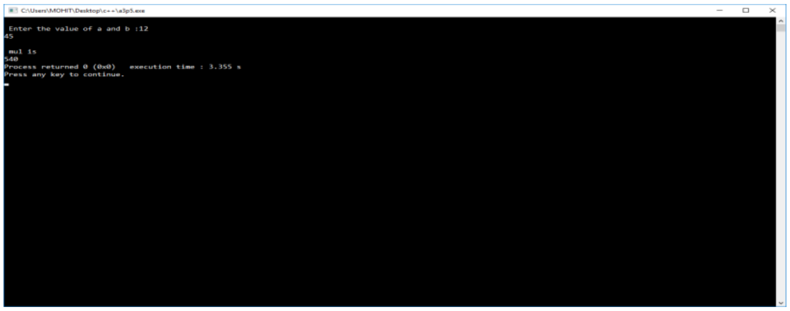
```
        a[i].getdata();
}

for(i=0;i<5;i++)
{
        a[i].putdata();
}

cout<<"\n\n\nYou want to find any record ? : ";
cin>>c;

if(c=='y' || c=='Y')
{
        cout<<"Enter account id. :";
        cin>>data;

        for(i=0;i<5;i++)
        {
          flag=a[i].search(data);

          if(flag==1)
          {
                break;
          }
        }

        if(flag==0)
```

```
            {
                cout<<"\n\n***********\nAccount is not found";
            }
        }
        else
        {
         exit(0);
        }

        return 0;

}
```



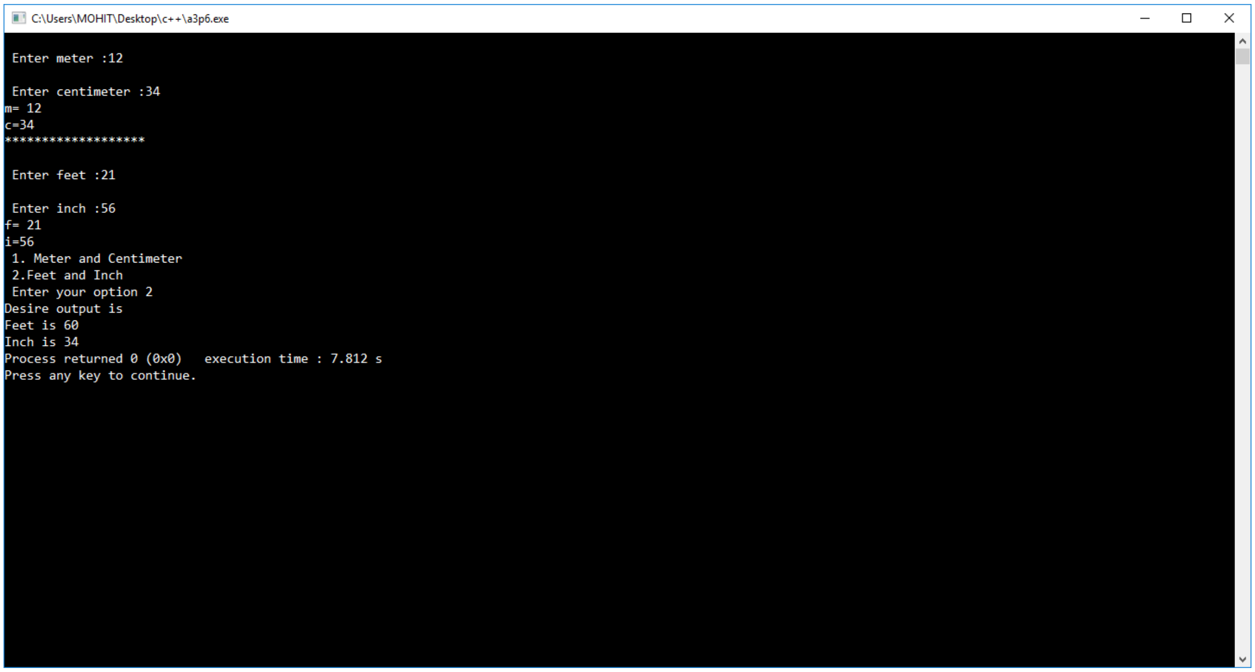
```
cust_name : mohit
account_id : 1
amount: 23

cust_name : chand
account_id : 2
amount: 345

cust_name : parth
account_id : 3
amount: 3456

cust_name : meet
account_id : 4
amount: 467

cust_name : nand
account_id : 5
amount: 255



You want to find any record ? : y
Enter account id. :2



***********
Account is found

Custumer name :chand
Account id    :2
Act balance   :345

Process returned 0 (0x0)   execution time : 37.480 s
Press any key to continue.
```

```
// programme 5
/*Create two classes X and Y containing private variables x and y
respectively. Using a common friend function, perform multiplication
operation between x and y.
*/
#include<stdio.h>
#include<iostream>
using namespace std;
class Y;
class X{
   private:
      int x;
   public:
      void setdata(int);
      friend void mul(X,Y);
};
class Y{
private:
   int y;
public:
   void setdata(int);
   friend void mul(X,Y);

};
void X::setdata(int a)
{
        x=a;
}
void Y::setdata(int b)
{
    y=b;
}
void mul(X o1,Y o2)
{

    cout << "mul is \n" << o1.x*o2.y;
}
int main()
{
    int a,b;
    cin >> a >> b;
    X obj1;
    Y obj2;
    obj1.setdata(a);
    obj2.setdata(b);
    mul(obj1,obj2);
    return 0;
}
```



```
 Enter the value of a and b :12
45

 mul is
540
Process returned 0 (0x0)   execution time : 3.355 s
Press any key to continue.
```

```
//programme 6

/*"Create two classes DM and DB which store the value of distances.
DM stores distances in meters and centimeters and DB in feet and
inches.

Write a program that can read values for the class objects and add one

object of DM with another object of DB. Use a friend function to carry
out the

addition operation. The object stores the results may a DM object or DB

object, depending on the units in which the results are required. The
display

should be in the format of feet and inches or meters and centimeters
depending on the object on display.

1 Feet = 0.3048Meter        1Meter = 3.28 Feet

1 Inch = 2.54 Centimeter     1 Centimeter = 0.3937 Inch" */


#include<iostream>

using namespace std;

class DB;

class DM{

    int meter,centi;

  public:

      void getdata(DM);

      void putdata(DM);

    friend void addition(DM &,DB &);

};

class DB {
        int feet,inche;
    public:
        void getdata(DB);
        void putdata(DB);
        friend void addition(DM &,DB &);
};
void DM::getdata(DM b)
{
    cout << "\n Enter meter :" ;
            cin >> meter;
    cout << "\n Enter centimeter :";
            cin >> centi;
}
void DM::putdata(DM b)
{
        cout << "m= " << meter;
        cout << "\nc=" << centi;
}
void DB::getdata(DB m)
{
    cout << "\n Enter feet :" ;
            cin >> feet;
    cout << "\n Enter inch :";
            cin >> inche;
}
void DB::putdata(DB m)
{
    cout << "f= " << feet;
    cout << "\ni=" << inche;
}
void addition(DM &x,DB &y)
{
  int option,ans1,ans2;
      cout << "\n 1. Meter and Centimeter ";
      cout << "\n 2.Feet and Inch " ;
      cout << "\n Enter your option ";
            cin >> option;

    switch(option){
        case 1:
          ans1=x.meter+y.feet*0.3048;
          ans2=x.centi+y.inche*2.54;
              cout << "Desire output is \n";
              cout << "Meter is " << ans1 << endl;
              cout << "Centimeter is " << ans2 ;
              break;
        case 2:
          ans1=x.meter*3.28+y.feet;
          ans2=x.centi*0.3937+y.feet;
              cout << "Desire output is \n";
              cout << "Feet is " << ans1 << endl;
              cout << "Inch is " << ans2 ;
              break;
        case 3:
          cout << "\n Invalid option ";
          break;
       }
}
int main()
{
    DM x;
    DB y;
    x.getdata(x);
    x.putdata(x);
      cout << "\n*******************\n";
    y.getdata(y);
    y.putdata(y)
    addition(x,y);
        return 0;}
```

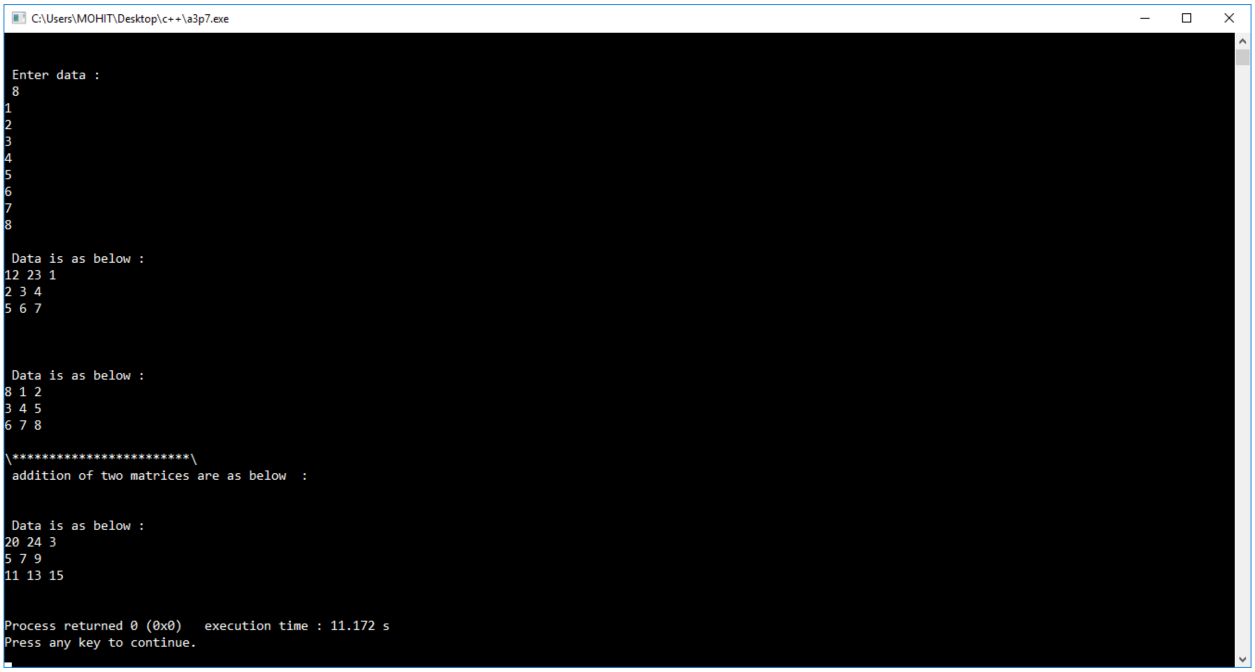


```

 Enter meter :12

 Enter centimeter :34
m= 12
c=34
*******************

 Enter feet :21

 Enter inch :56
f= 21
i=56
 1. Meter and Centimeter
 2.Feet and Inch
 Enter your option 2
Desire output is
Feet is 60
Inch is 34
Process returned 0 (0x0)   execution time : 7.812 s
Press any key to continue.
```

```
//programme 7
/* Define a class matrix with an integer array of 3X3 as a data member.
Define a friend function which adds two matrix objects and returns
resultant matrix object.
*/
#include<iostream>
#include<iomanip>
using namespace std;
class matrix{
      int arr[3][3];
public:
   void getdata(matrix);
   void putdata(matrix);
   friend matrix sum(matrix &,matrix &);
};
void matrix::getdata(matrix){
      cout << "\n Enter data :  \n ";
         for(int i=0;i<3;i++)
         {
            for(int j=0;j<3;j++)
            {
               cin >> arr[i][j];
            }
         }
```

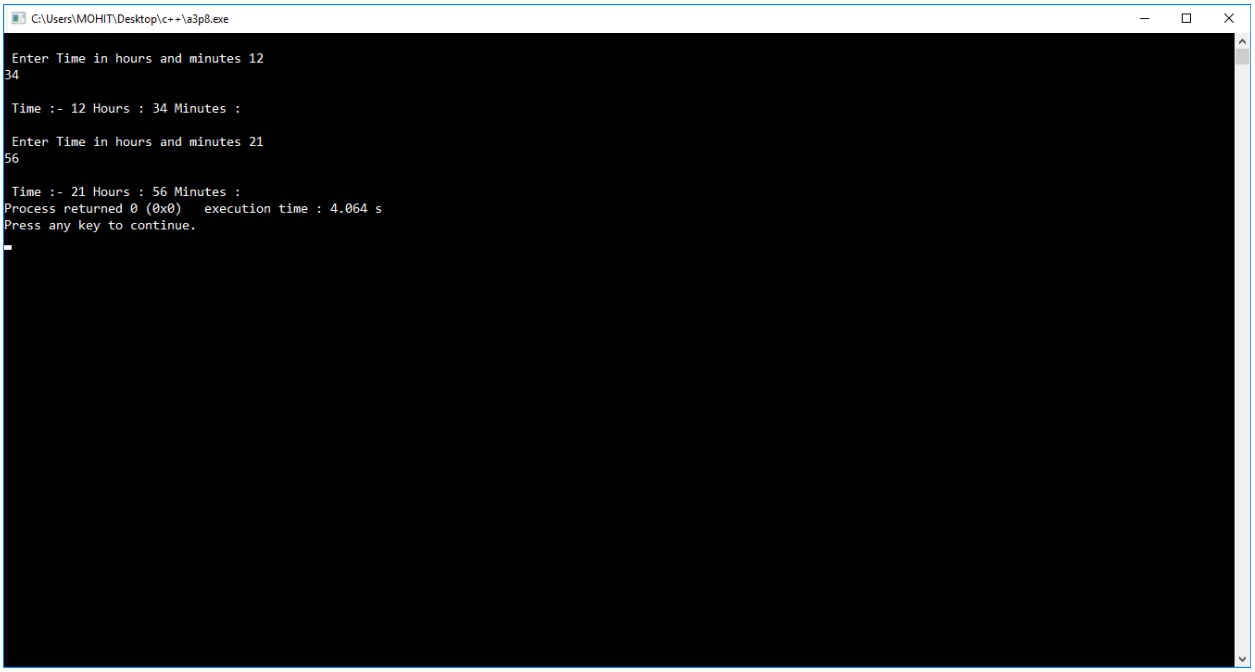
```
}
void matrix::putdata(matrix){
cout << "\n Data is as below : \n" ;
    for(int i=0;i<3;i++)
       {
          for(int j=0;j<3;j++)
          {
             cout << setw(-5) << arr[i][j] << " ";
          }
                cout << endl;
       }
       cout << endl;
}
matrix sum(matrix &p,matrix &q){
    matrix r;
    for(int i=0;i<3;i++)
    {
       for(int j=0;j<3;j++)
       {
          r.arr[i][j]=p.arr[i][j]+q.arr[i][j];
       }
    }
    return r;
}
int main()
{
    matrix a,b,c;
```

```
        a.getdata(a);
          a.putdata(a);
            cout << endl;
        b.getdata(b);
          b.putdata(b);
      cout << "\\************************\\";
        c=sum(a,b);
      cout << "\n addition of two matrices are as below  : \n\n ";
          c.putdata(c);
          return 0;
}
```



```

 Enter data :
 8
1
2
3
4
5
6
7
8

 Data is as below :
12 23 1
2 3 4
5 6 7



 Data is as below :
8 1 2
3 4 5
6 7 8

\************************\
 addition of two matrices are as below  :


 Data is as below :
20 24 3
5 7 9
11 13 15


Process returned 0 (0x0)   execution time : 11.172 s
Press any key to continue.
```

```
  // programme 8

 /* "Define a class Time with hours and minutes as two data members, add
necessary member functions to initialize and display data of class. Do not use
constructors in a class. Define a member function sum() which adds two Time
objects. Invoke the statements like T3.sum(T1, T2) in main()."
*/
#include<iostream>
using namespace std;
class time{
    public:
  int hours;
  int minutes;
    public:
        void getdata(time);
        void putdata(time);
        void sum(time,time);
};
 void time::getdata(time)
 {
     cout << "\n Enter Time in hours and minutes ";
     cin >> hours >> minutes;
 }
 void time ::putdata(time)
 {
     cout << "\n Time :- " << hours << " Hours : " << minutes << " Minutes : ";
 }
 void time::sum(time T1,time T2)
{
    time T3;
    T3.minutes=T1.minutes+T2.minutes;
    T3.hours=T3.minutes/60;
    T3.minutes=T3.minutes%60;
    T3.hours+=T1.hours+T2.hours;
    cout<<"****************************";
    cout<<"Your Answer :\n";
    T3.putdata(T3);
    cout<<"\n****************************";
 }
int main()
 time T3,T1,T2;
        T1.getdata(T1);
        T1.putdata(T1);
            cout << endl;
        T2.getdata(T2);
        T2.putdata(T2);
        return 0;
}
```



```
 Enter Time in hours and minutes 12
34

 Time :- 12 Hours : 34 Minutes :

 Enter Time in hours and minutes 21
56

 Time :- 21 Hours : 56 Minutes :
Process returned 0 (0x0)   execution time : 4.064 s
Press any key to continue.
```