|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| /\*Create two class DM and DB which store the value distances. DM stores distances in metres and centimetres 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 that stores the results maybe a DM object or DB object, depending on the units in which the results required. The display should be in the format of  feet and inches or metres and centimetres depending on the object on display.\*/  //Falgun Kole 22Co14 29-9-2023  #include<iostream>  #include<stdlib.h>  using namespace std;  class DB;  class DM  {   public:   int mts,cms;   DM()      {          mts=0;          cms=0;      }   DM(int m,int c)   {    mts=m;    cms=c;   }   void display()   {     cout<<"Distance is : "<< mts <<" meters and "<< cms <<" centimeters "<<endl;   }    friend void add(DM &m1, DB &b1);  };  class DB  {   public:   int feet,inch;   DB()      {          feet=0;          inch=0;      }   DB(int f,int i)   {    feet=f;    inch=i;   }    void display()   {     cout<<"Distance is : "<< feet <<" feet and "<< inch <<" inches "<<endl;   }   friend void add(DM &m1, DB &b1);  };  void add(DM & m1,DB & b1,int choice)  {   while(1)   {   float meters,centimeter,ft,in;      switch(choice){      case 1:cout<<"\*\*\*\*MTS  &&  CMS\*\*\*\*"<<endl;             centimeter=(m1.mts+b1.feet\*0.3048)\*100+b1.inch\*2.54;             m1.mts=centimeter/100;             m1.cms=centimeter-m1.mts\*100;             m1.display();             return;             break;      case 2:cout<<"\*\*\*\*FEET  &&  INCH\*\*\*\*"<<endl;             in=(b1.feet+m1.mts/0.3048)\*12+m1.cms/2.54;             b1.feet=in/12;             b1.inch=in-b1.feet\*12;             b1.display();             return;             break;      case 3:exit(0);             break;      default:cout<<"invalid input"<<endl;      return ;   }  }  }  int main()  {   int m,c,f,i,choice;      cout<<"Enter the values in meters and centimeters: "<<endl;      cin>>m>>c;      DM m1(m,c);      cout<<"Enter the values in feet and inches: "<<endl;      cin>>f>>i;      DB b1(f,i);      cout<<"Enter your choice"<<endl;      cin>>choice;      add(m1,b1,choice);   return 0;  }  Output:   |  | | --- | | Enter the values in meters and centimeters:  100  20  Enter the values in feet and inches:  120  20  Enter your choice  1  \*\*\*\*MTS && CMS\*\*\*\*  Distance is : 137 meters and 8 centimeters |   /\*Write a program to design a class complex to represent complex numbers. The complex class should use an external function (use it as friend function) to add  two complex numbers. The function should return an object of type complex representing the sum of two complex numbers.\*/  //Falgun Kole 22Co14 29-9-2023  #include<iostream>  using namespace std;  class Complex  {  public:      int real,image;      Complex()      {       real=0;       image=0;      }      Complex(int r,int i)      {       real=r;       image=i;      }      void display()      {       cout<<"sum of complex number is : "<<real<<" + "<<image<<"i"<<endl;      }      friend Complex add(Complex & C1,Complex & C2);  };  Complex add(Complex & C1,Complex & C2)  {   Complex temp;   temp.real = C1.real + C2.real;   temp.image = C1.image + C2.image;   return temp;  }  int main()  {   int real,image;   Complex C1(real,image);   Complex C2(real,image);   Complex C3;   cout<<"Enter the real and imaginary part for 1 : "<<endl;   cin>>C1.real>>C1.image;   cout<<"Enter the real and imaginary part for 2 : "<<endl;   cin>>C2.real>>C2.image;   C3 =add(C1,C2);   C3.display();   return 0;  }  Output:   |  | | --- | | Enter the real and imaginary part for 1 :  1  32  Enter the real and imaginary part for 2 :  12  10  sum of complex number is : 13 + 42i | |  |   /\*Kristen is a contender for gold medal of her high school. She wants to know how many students (if any) have scored higher than her in the exams given during  this semester. Create a class named Student with the following specifications An variable named scores to hold a student's 5 exam scores. A void input()  function that reads integers and saves them to. An int calculateTotalScore() friend function that returns the sum of the student's scores.  Read 5 scores for every student & save them to scores. Display how many students have got score higher than Kristen.  Sample Input  3  30 40 45 10 10  40 40 40 10 10  50 20 30 10 10  The first line contains , the number of students in Kristen's class (including  Kristen). The second line contains Kristens marks. The subsequent lines contain  each student's exam grades for this semester.  Sample output  1  Only student scored higher than her.\*/  //Falgun Kole 22Co14 29-9-2023  #include<iostream>  using namespace std;  class student  {   public:      int scores;      void input()      {          int i;          cout<<"Enter the marks of 5 subjects: "<<endl;          for(i=0;i<5;i++)          {              cin>>scores;          }      }      friend int calculateTotal(student s);  };  int calculateTotal(student s)  {     int sum=0,i;     for(i=0;i<5;i++)     {         sum=sum+s.scores;     }     return sum;  }  int main()  {      int noofstudents,count=0,i;      cout<<"Enter the number of students: "<<endl;      cin>>noofstudents;      student s[noofstudents];      cout<<"Enter Kristen's marks: "<<endl;      s[0].input();      for(i=1;i<noofstudents;i++)      {          cout<<"Enter marks of student No."<<i+1<<endl;          s[i].input();      }      for(i=1;i<noofstudents;i++)      {          if(calculateTotal (s[0])<calculateTotal (s[i]))          {              count++;          }      }      cout<<"Total number of students who have scored more than Kristen is: "<<count<<endl;  }  Output:   |  | | --- | | Enter the number of students:  2  Enter Kristen's marks:  Enter the marks of 5 subjects:  92  89  90  28  90  Enter marks of student No.2  Enter the marks of 5 subjects:  80  43  90  99  100  Total number of students who have scored more than Kristen is: 1 | |