**CODE:-**

# include <iostream>

# include <iomanip>

using namespace std;

class complex

{

float real,imag;

public:

void display()

{

cout<<"The complex number is : "<< setprecision(4)<<real<<" + "<<setprecision(4)<<imag<<"I"<<endl;

}

void enter()

{

cout<<"Enter the real and imaginary parts."<<endl;

cin>>real>>imag;

}

friend complex sum(complex &c1,complex &c2);

};

complex sum(complex &a, complex &b)

{

complex c;

c.real=a.real+b.real;

c.imag=a.imag+b.imag;

return c;

}

int main()

{

complex c1,c2,c3;

c1.enter();

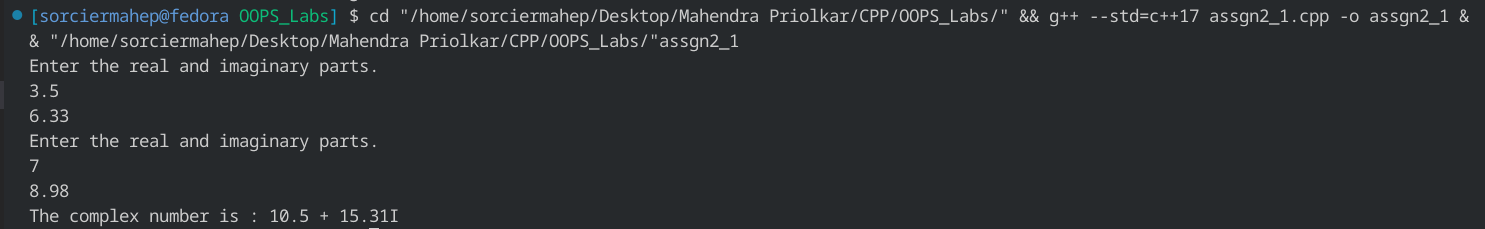
c2.enter();

c3=sum(c1,c2);

c3.display();

}

**OUTPUT:-**

****

**CODE:-**

# include <iostream>

using namespace std;

class profile

{

char user[30];

int units;

float price=0.0;

public:

void enter()

{

cout<<"Enter name."<<endl;

cin>>user;

cout<<"Enter units."<<endl;

cin>>units;

}

void calc()

{

while(units>0)

{

if(units<=100)

{

price+=units\*0.6;

units=0;

}

if(units>100 && units <=300)

{

price+=(units-100)\*0.8;

units=100;

}

if(units>300)

{

price+=(units-300)\*0.9;

units=300;

}

}

if(price<50.0)

price=50.0;

if(price>300.0)

price+=0.15\*price;

}

void display()

{

cout<<"The charge incurred by user "<<user<<" is "<<price<<" Rs."<<endl;

}

};

int main()

{

int num;

cout<<"Enter the total number of users."<<endl;

cin>>num;

profile p[num];

for(int i=0;i<num;i++)

{

cout<<"Enter details of user "<<i+1<<"."<<endl;

p[i].enter();

p[i].calc();

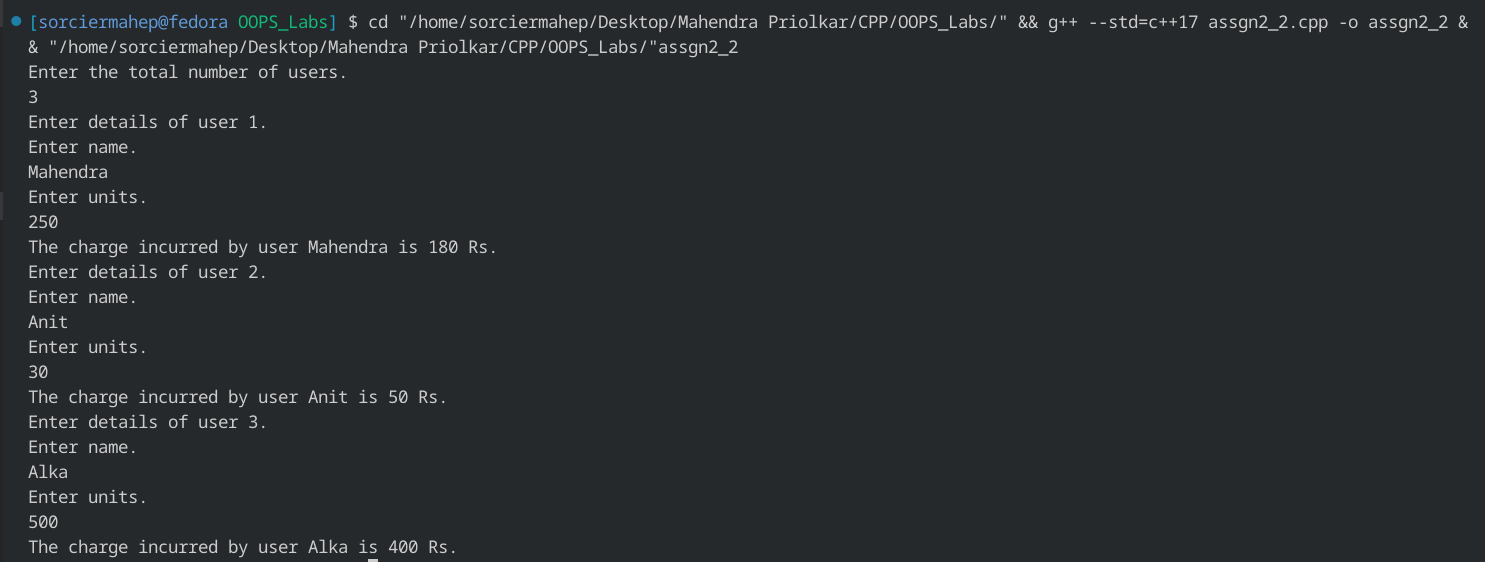
p[i].display();

}

return 0;

}

**OUTPUT:-**



**CODE:-**

#include <iostream>

using namespace std;

class DB;

class DM

{

int m, cm;

public:

DM()

{

m = 0.0;

cm = 0.0;

}

void get()

{

cout << "Enter the data in metric units." << endl;

cin >> m >> cm;

}

void display()

{

cout << "\nMetre: " << m;

cout << "\nCentimetre: " << cm<<endl;

}

friend void add(DM, DB, int);

};

class DB

{

int ft, ic;

public:

DB()

{

ft = 0.0;

ic = 0.0;

}

void get()

{

cout << "Enter the data in imperial units." << endl;

cin >> ft >> ic;

}

void display()

{

cout << "\nFeet: " << ft;

cout << "\nInch: " << ic<<endl;

}

friend void add(DM, DB, int);

};

void add(DM a, DB b, int n)

{

if (n == 1)

{

DM sum;

float c = (a.m \* 100.0 + a.cm + b.ft \* 30.48 + b.ic \* 2.54);

if (c >= 100)

{

sum.m = c / 100;

sum.cm = c - sum.m \* 100;

}

else

{

sum.m = 0.0;

sum.cm = c;

}

sum.display();

}

else if (n == 2)

{

DB sum;

float i = (a.m \* 39.37 + a.cm / 2.54 + b.ft \* 12.0 + b.ic);

if (i >= 12)

{

sum.ft = i / 12;

sum.ic = i - sum.ft \* 12;

}

else

{

sum.ft = 0.0;

sum.ic = i;

}

sum.display();

}

}

int main()

{

DM a;

DB b;

int ch;

a.get();

b.get();

cout << "Enter 1 for result in metric units or 2 for result in imperial units." << endl;

cin >> ch;

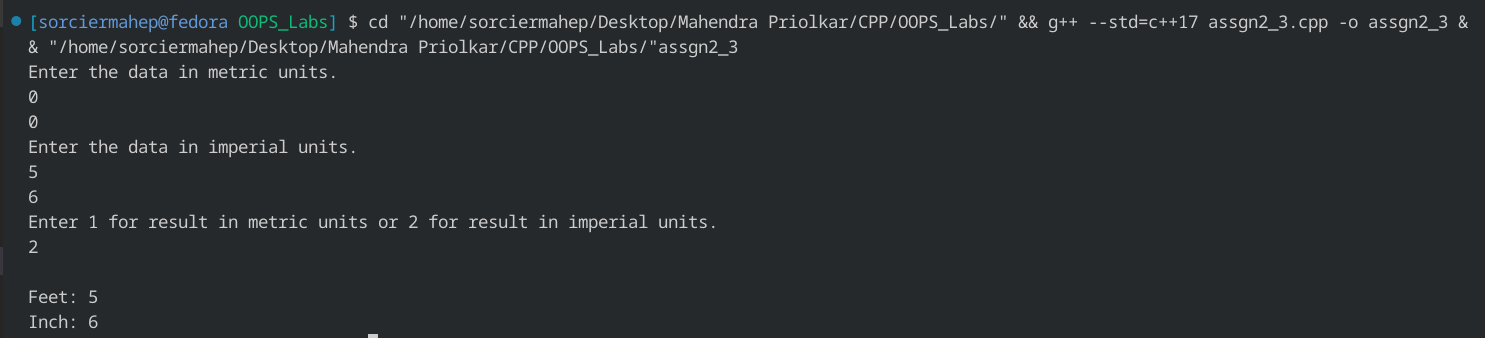
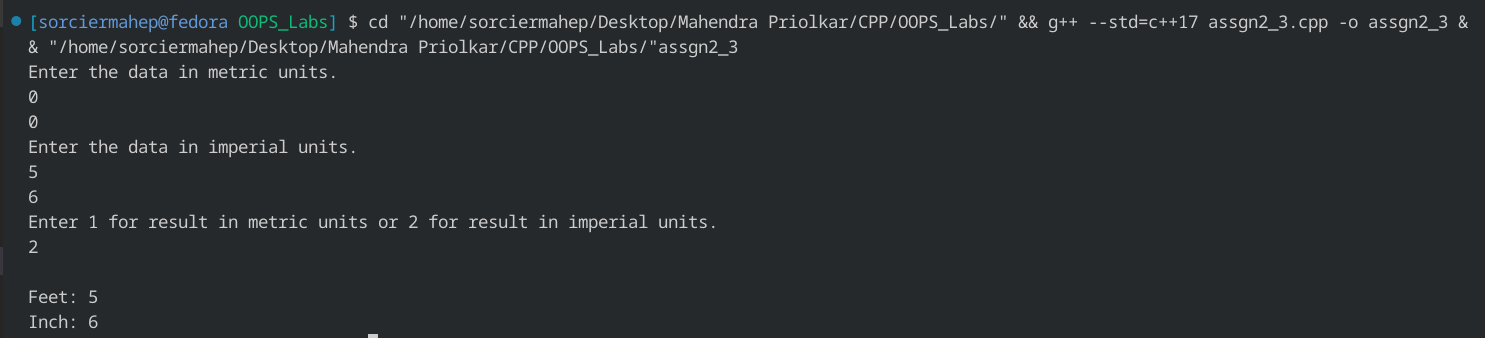
if (ch == 1 || ch == 2)

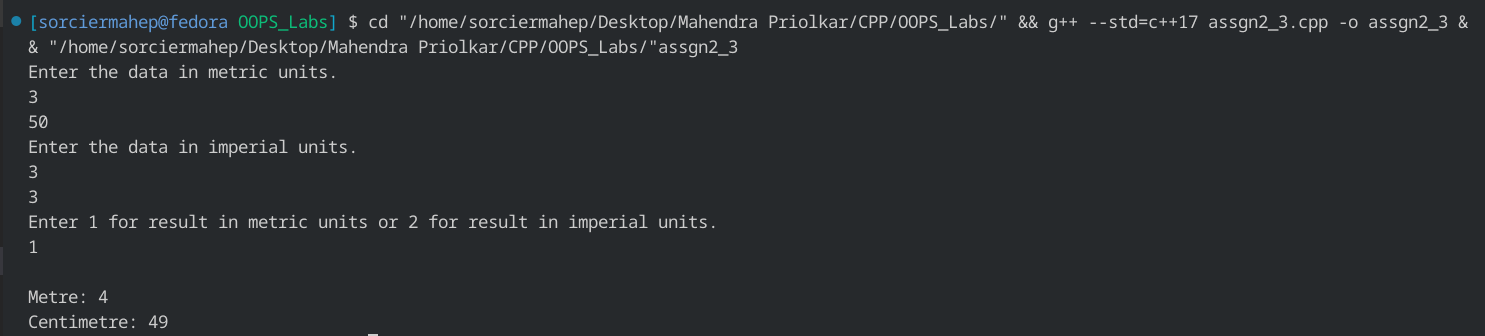
add(a, b, ch);

return 0;

}

**OUTPUT:-**





**CODE:-**

#include <iostream>

using namespace std;

class counting

{

int count[6];

public:

counting()

{

for (int i = 0; i < 6; i++)

count[i] = 0;

}

void vote(int n)

{

count[n]++;

}

void display()

{

for (int i = 0; i < 5; i++)

cout << "Candidate " << i + 1 << " : " << count[i] << " votes." << endl;

cout << "Spoilt ballots: " << count[5]<<endl;

}

};

int main()

{

counting c;

int ch;

cout << "Enter votes of candidates from 1 to 5." << endl;

cout << "Enter 0 to exit and display result." << endl;

cout << "Other entries will be counted as spoiled." << endl;

while (1)

{

cin >> ch;

if (ch >= 1 && ch <= 5)

c.vote(ch - 1);

else if (ch != 0)

c.vote(5);

else

{

c.display();

break;

}

}

}

**OUTPUT:-**

