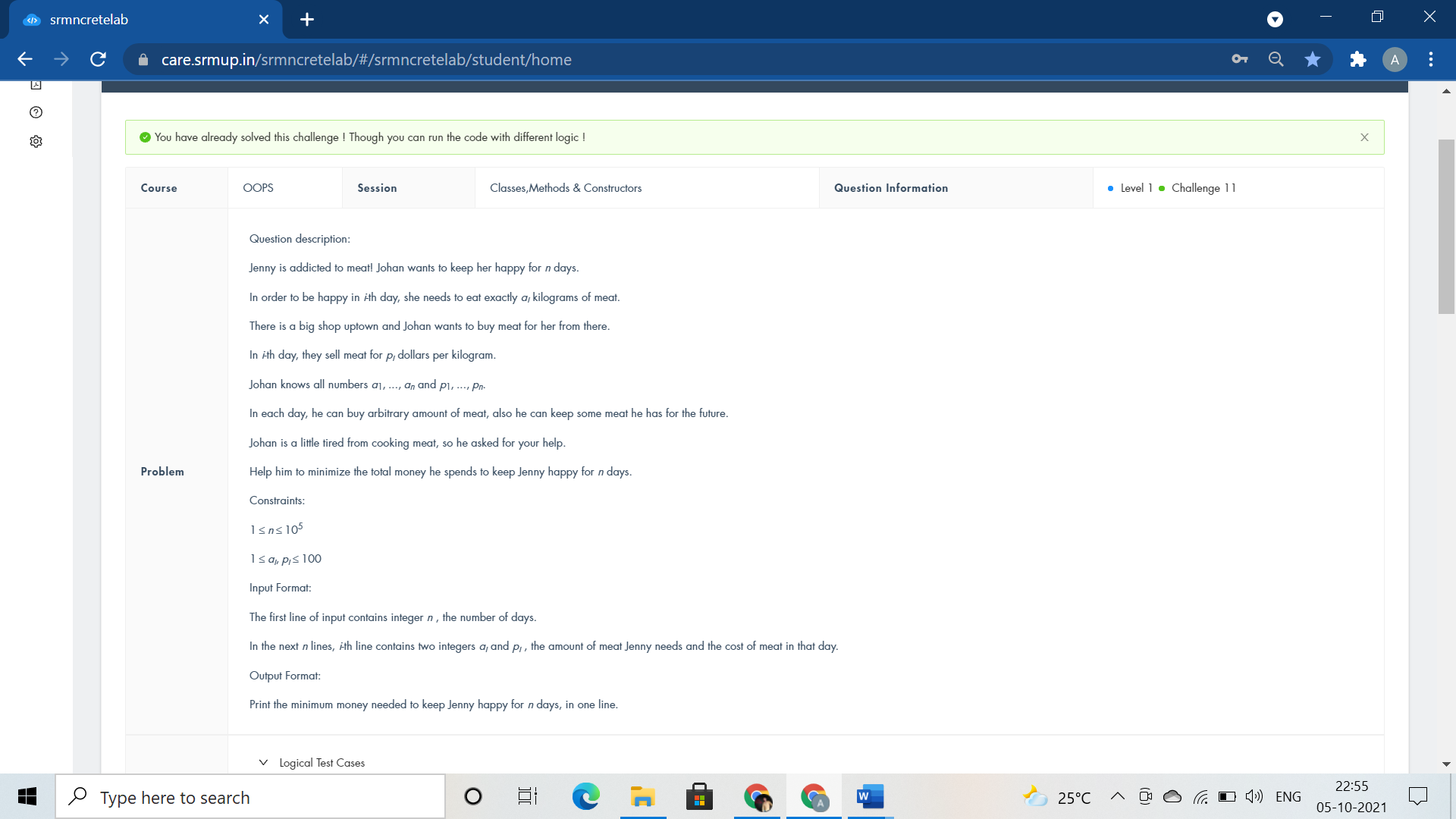
Level 1 Qns Topic:- Class Methods and Constructors



#include <iostream>

using namespace std;

class Happiness{

public:int Meat(){

int n,a,b,max=100,sum=0;

cin>>n;

while(n--)

{

cin>>a>>b;

//max=b;

if(b>=max)

sum+=a\*max;

// cout<<max<<endl;

// cout<<sum<<endl;

else

{

max=b;

sum+=a\*b;

// cout<<max<<endl;

// cout<<sum<<endl;

}

}

return sum;

}

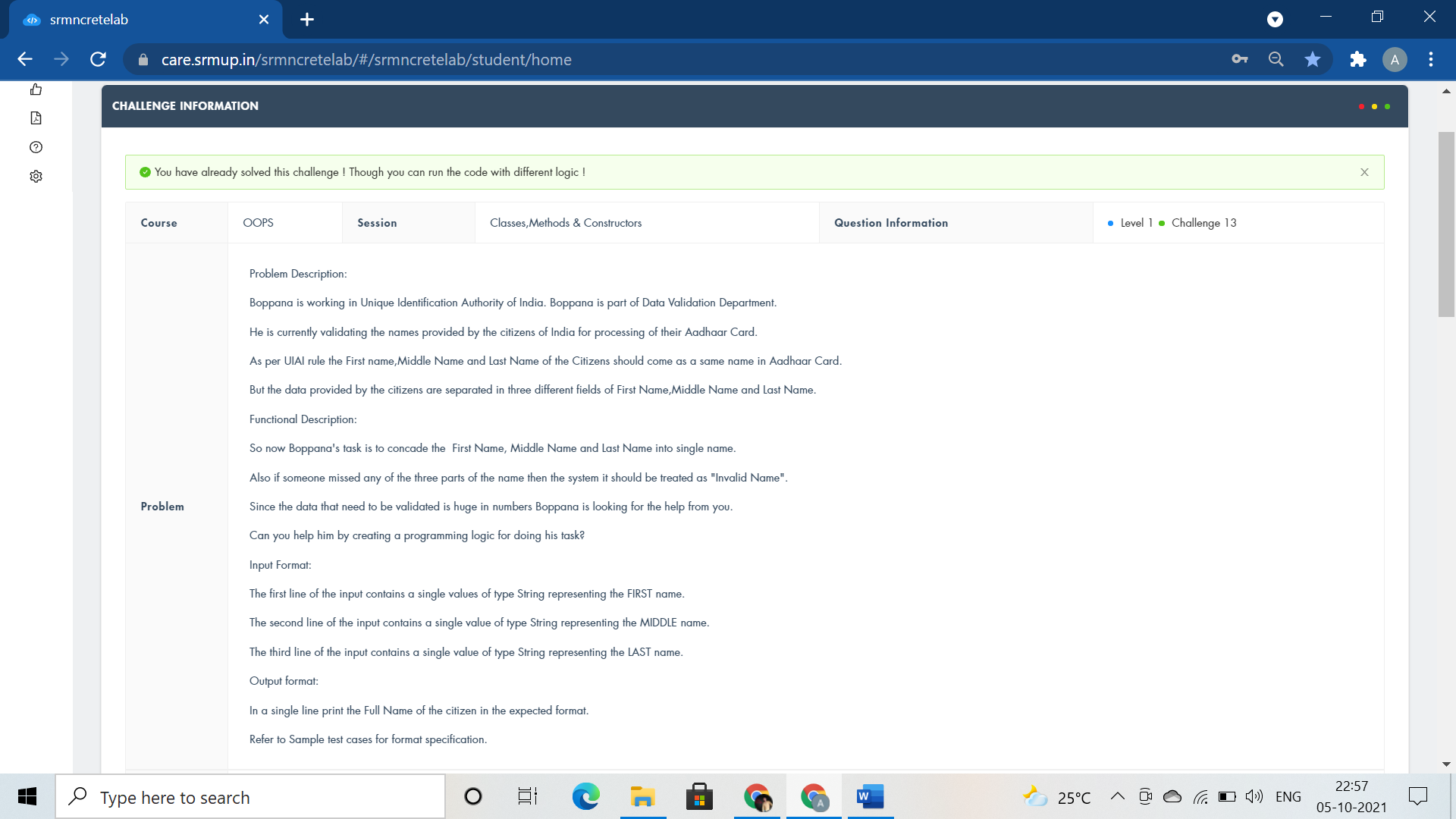
};

int main(){

Happiness Purchase;

cout<<Purchase.Meat();

}



#include <iostream>

#include<cstring>

#include<string>

using namespace std;

class aadhaar

{

public:

void NameofCitizen(string fn,string mn,string ln)

{

if(fn.empty() || mn.empty() || ln.empty() )

{

cout<<"Invalid Name";

}

//cout<<"Invalid name"; exit(0) :

else

cout<<fn<<mn<<ln;

}

};

int main()

{

aadhaar Card;

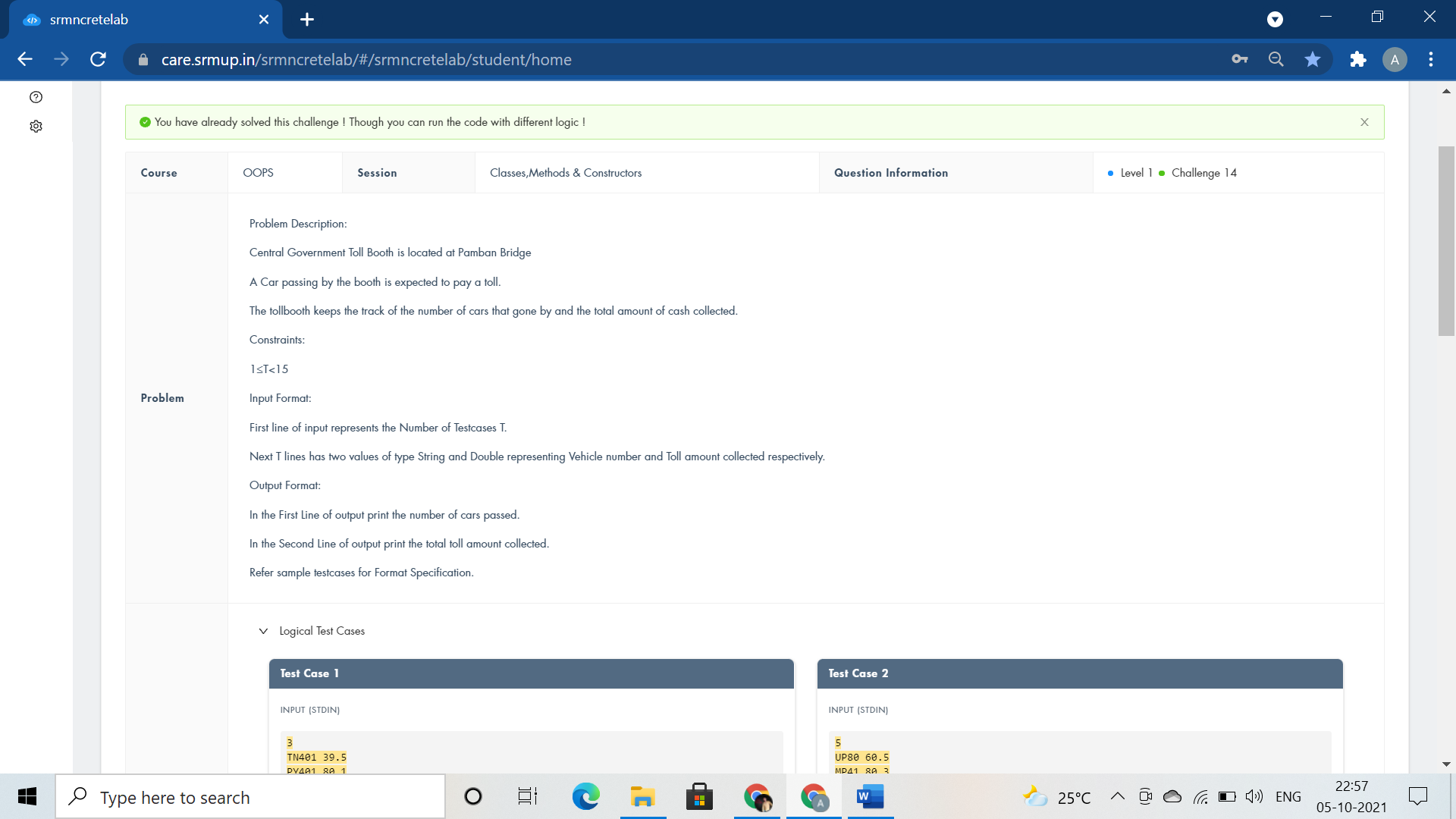
string fn,mn,ln;

cin>>fn>>mn>>ln;

Card.NameofCitizen(fn,mn,ln);

return 0;

}



#include <iostream>

using namespace std;

class TollBooth

{

public:

int cars;

float tollcollected;

TollBooth(){

cars=0;

tollcollected=0;

}

void payingcar(double pay){

cars++;

tollcollected+=pay;

}

void nonpayingcar(){

cars++;

}

void display(){

cout<<cars<<endl<<tollcollected<<endl;

}

};

int main()

{

TollBooth obj;

char VehicleNo[10];

float TollAmt;

int carpassed,i;

cin>>carpassed;

for(i=0;i<carpassed;i++)

{

cin>>VehicleNo>>TollAmt;

if(TollAmt>0) obj.payingcar(TollAmt);

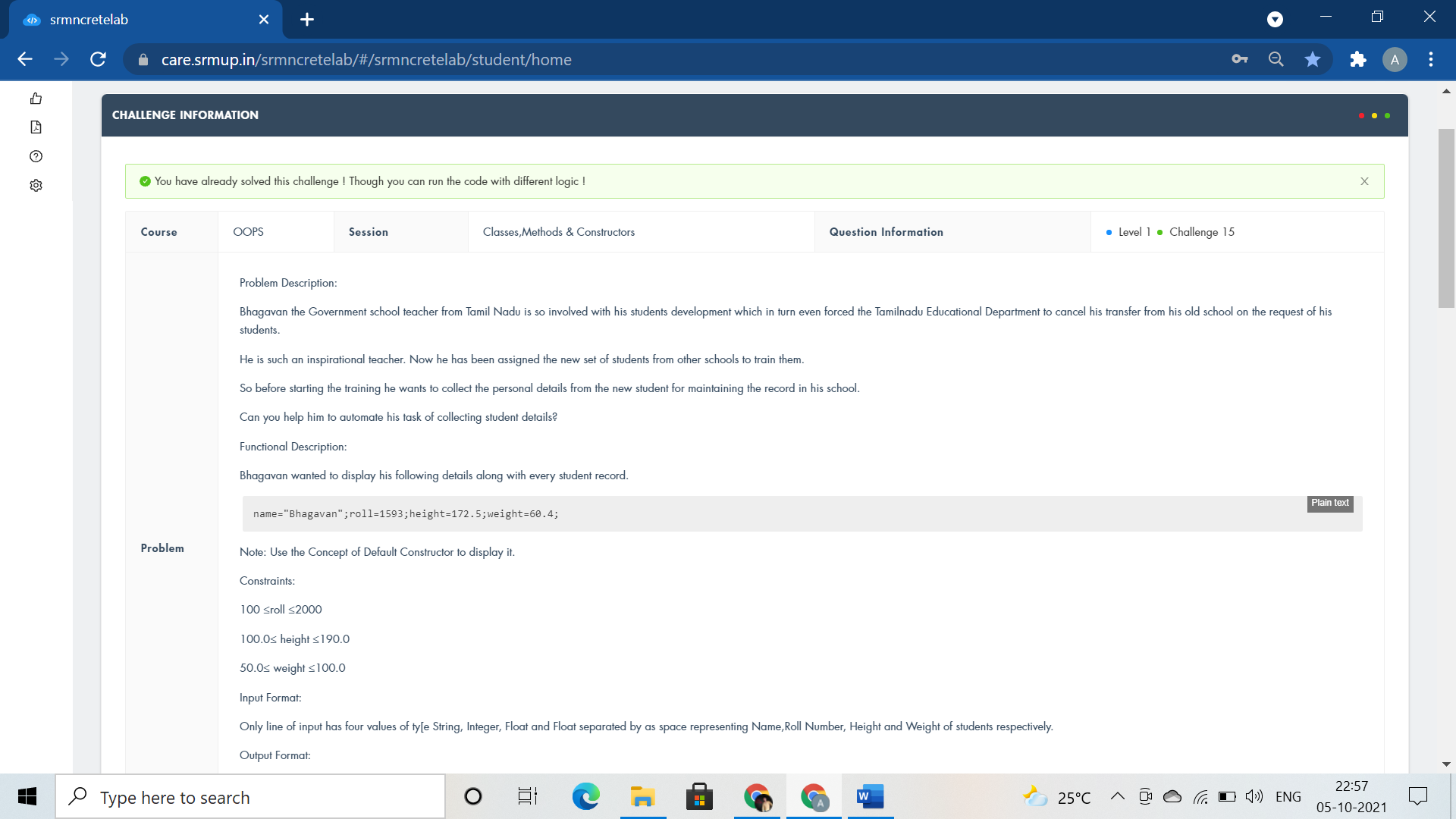
else obj.nonpayingcar();

}

obj.display();

return 0;

}



#include <bits/stdc++.h>

//#include<iomanip>

//#include<string>

using namespace std;

class student

{

string name;

int roll;

float height,weight;

public:

student(){name="Bhagavan";roll=1593;height=172.5;weight=60.4;}

void getdata() {

cin>>name>>roll>>height>>weight;

}

void displaydata(){

cout<<name<<" "<<roll<<" "<<height<<" "<<weight<<endl;

}

};

int main()

{

student s1,s2;

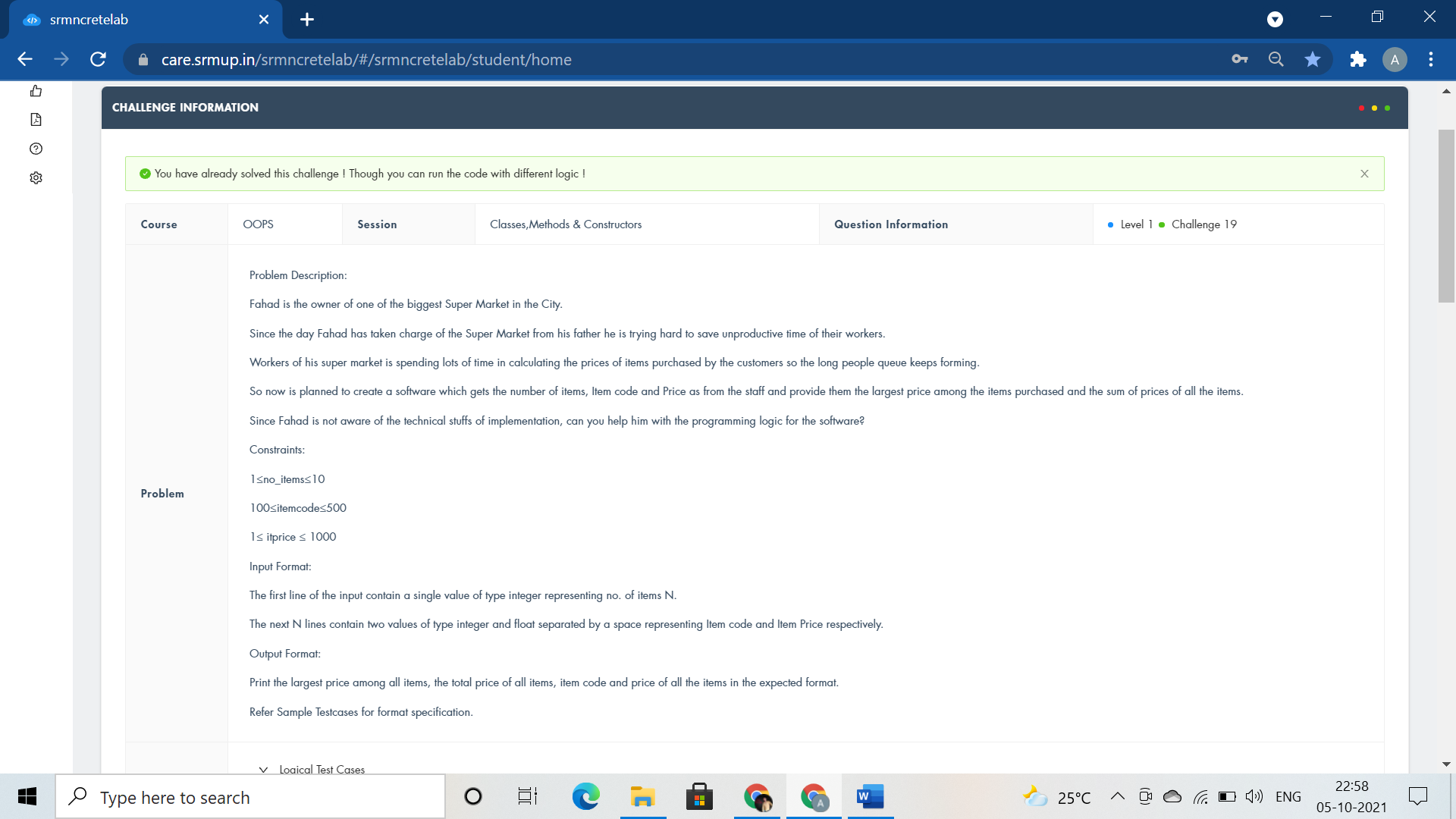
s1.getdata();

s1.displaydata();

s2.displaydata();

return 0;

}



#include <iostream>

using namespace std;

class ITEM

{

public:

int n;

float large=0,summ=0;

float arr[100],code[100];

void getdata(int b){

n=b;

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

cin>>code[i]>>arr[i];

}

void largest(){

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

{

if(arr[i]>=large)

large=arr[i];

}

}

void sum(){

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

summ+=arr[i];

}

void displayitems(){

cout<<"Largest Price="<<large<<endl;

cout<<"Sum of Prices="<<summ<<endl;

cout<<"Code and Price"<<endl;

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

cout<<code[i]<<" and "<<arr[i]<<endl;

}

};

using namespace std;

int main()

{

ITEM order;

int b;

cin>>b;

order.getdata(b);

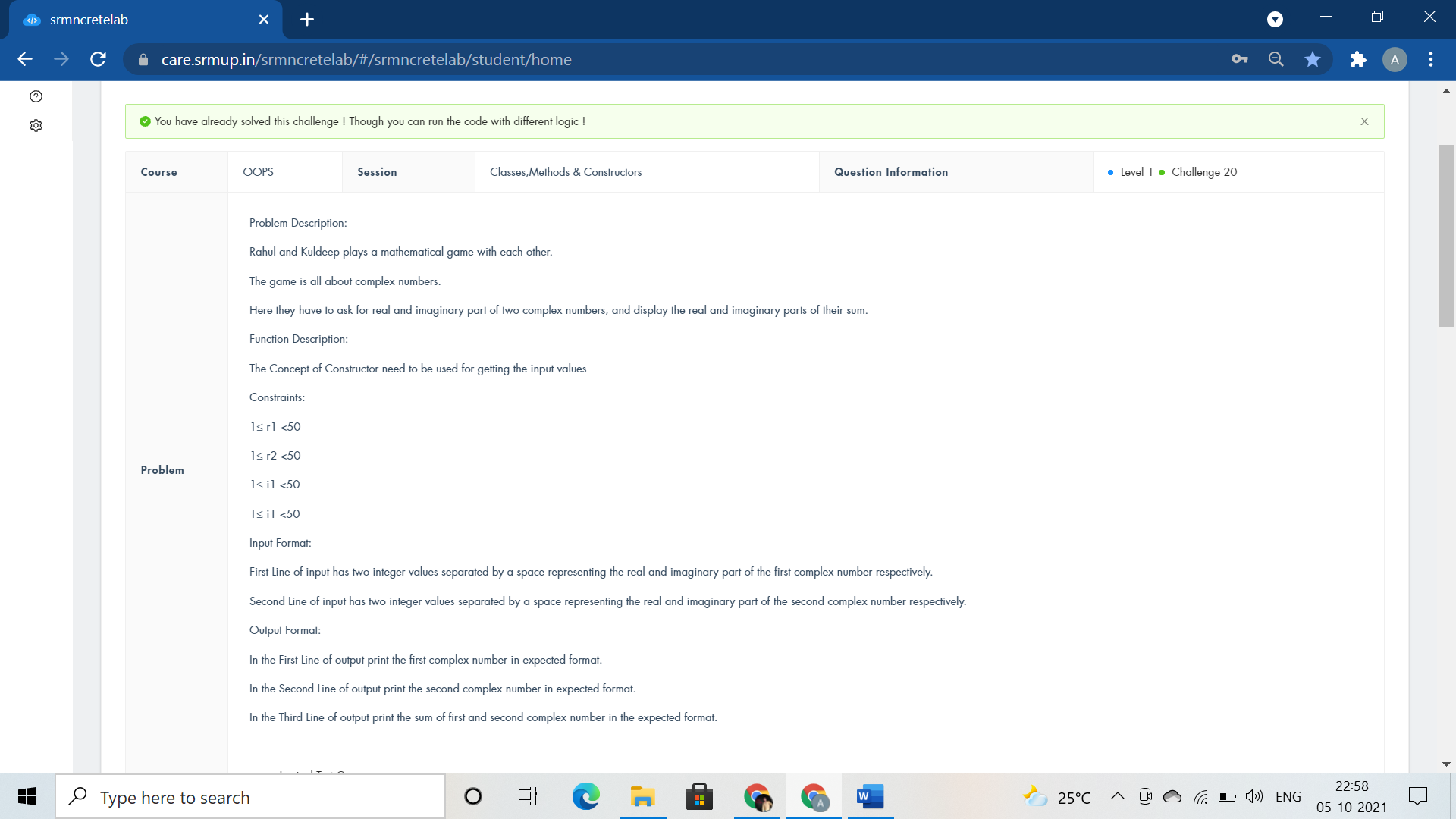
order.largest();

order.sum();

order.displayitems();

return 0;

}



#include<iostream>

using namespace std;

class Complex{

public:

int r1,i1,r2,i2,r3,i3;

Complex(){cin>>r1>>i1;cin>>r2>>i2;}

void addcomplex(){

r3=r1+r2;

i3=i1+i2;

}

void displaycomplex(){

cout<<r1<<"+"<<i1<<"i"<<endl;

cout<<r2<<"+"<<i2<<"i"<<endl;

cout<<r3<<"+"<<i3<<"i"<<endl;

}

};

int main(){

Complex calculate;

calculate.addcomplex();

calculate.displaycomplex();

return 0;

}

Graphical user interface, text, application, email

Description automatically generated

#include <iostream>

#include <string.h>

#include <stdio.h>

using namespace std;

double a[18][18], b[1 << 18];

int fun(int x) {

int s = 0;

while (x)

{

s += x & 1;

x >>= 1;

}

return s;

}

int main() {

if(0)

cout<<"class Lake public:void survival() fish.survival();";

int n, i, r, t, j;

cin >> n;

for (i = 0; i < n; i++)

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

scanf("%lf", &a[i][j]);

memset(b, 0, sizeof(b));

b[(1 << n) - 1] = 1;

for (i = (1 << n) - 1; i >= 0; i--) {

int c = fun(i);

c = c \* (c - 1) / 2;

for (r = 0; r < n; r++)

if (i & (1 << r))

for (t = 0; t < n; t++)

if (i & (1 << t))

b[i - (1 << t)] += b[i] \* a[r][t] / c;

}

for (r = 0; r < n - 1; r++)

printf("%.6lf ", b[1 << r]);

printf("%.6lf\n", b[1 << r]);

}

Graphical user interface, text, application, email

Description automatically generated

#include <iostream>

using namespace std;

class Friends

{

public:void Gifts(){

int i, n, a, b[50] = { 0 };

cin >> n;

for (i = 1; i < n+1; i++)

{

cin >> a;

b[a] = i;

}

for (i = 1; i < n+1; i++)

cout<< b[i]<<" ";

}

};

int main()

{

Friends Sharing;

Sharing.Gifts();

}

Graphical user interface, text, application, email, website

Description automatically generated

#include <iostream>

using namespace std;

class GoodNum

{

public:

void check(int tNum)

{

int cnt=0;

int rem;

while(tNum>0)

{

rem=tNum%10;

if(rem==0)

cnt++;

tNum/=10;

}

if(cnt==0)

cout<<"GOOD Number"<<endl;

else

cout<<cnt;

}

};

int main(){

int N;

cin>>N;

GoodNum Learning;

Learning.check(N);

return 0;

}

Graphical user interface, text, application, email

Description automatically generated

#include <iostream>

#include <math.h>

using namespace std;

class Building

{

public:

int length, width, ratePerSqFeet;

void calculateCost()

{

int i,j,k,z;

cin>>i>>j>>k;

length=i;

width=j;

ratePerSqFeet=k;

z=length\*width\*ratePerSqFeet;

cout<<"Cost of the Building : "<<z<<endl;

}

void determineSuitability()

{

if(length==70||length==410)

{

cout<<"Stability : Suitable";

}

else if(abs(length-width)<10)

{

cout<<"Stability : Suitable"<<endl;

}

else

{

cout<<"Stability : Not Suitable"<<endl;

}

}

};

int main()

{

Building construction;

construction.calculateCost();

construction.determineSuitability();

return 0;

}

Graphical user interface, text, application, email

Description automatically generated

#include <iostream>

using namespace std;

class address

{

int hno;

char cty[20];

char state[20];

public:

void getad()

{

cin>>hno>>cty>>state;

}

void putad()

{

cout<<"House No="<<hno<<endl;

cout<<"City="<<cty<<endl;

cout<<"State="<<state<<endl;

}

};

class house

{

char housename[30];

address a;

int n;

public:

void input();

};

void house::input()

{

cin>>housename;

cout<<"House name="<<housename<<endl;

a.getad();

a.putad();

cin>>n;

int lenght,widht,height;

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

{

cin>>lenght>>widht>>height;

cout<<"Detail of Room "<<i+1<<endl;

cout<<"Length="<<lenght<<endl;

cout<<"Breadth="<<widht<<endl;

cout<<"Height="<<height<<endl;

}

}

int main() {

if(0)

{

cout<<"void house::display()";

}

house x;

x.input();

return 0;

}

Graphical user interface, text

Description automatically generated

#include <iostream>

using namespace std;

class Bank

{

int total;

public:

void totalMoney(int n)

{

int r;

r = n%7;

n/=7;

total =(n\*(49+(7\*n)))/2 + r\*(2\*(n+1)+r-1)/2;

cout<<total;

}

};

int main(){

int n;

cin>>n;

Bank CalculateMoney;

CalculateMoney.totalMoney(n);

return 0;

}

Graphical user interface, text, application

Description automatically generated

#include <iostream>

using namespace std;

class student

{

string name;

int roll;

float height, weight;

public:

student(){name="Bhagavan";roll=1593;height=172.5;weight=60.4;}

void set\_data()

{

cin>>name>>roll>>height>>weight;

}

void displaydata()

{

cout<<name<<" "<<roll<<" "<<height<<" "<<weight<<endl;

}

};

int main()

{

student s1,s2;

s1.set\_data();

s1.displaydata();

s2.displaydata();

return 0;

}

Graphical user interface, text, application

Description automatically generated

#include <iostream>

using namespace std;

class Phone

{

public:

char n[14];

void change()

{

cin>>n;

n[0]='1';

cout<<'9'<<n;

}

};

int main()

{

Phone obj;

obj.change();

return 0;

}