

#include <bits/stdc++.h>

using namespace std;

template <class Forest>

Forest Visit(Forest a,Forest b){

if(a>b)

cout<<"Kayal\n";

else

cout<<"Elavenil\n";

return 1;

}

int main()

{

int a,b;

cin>>a>>b;

if(a%(a-b)==0 && b%(a-b)==0)

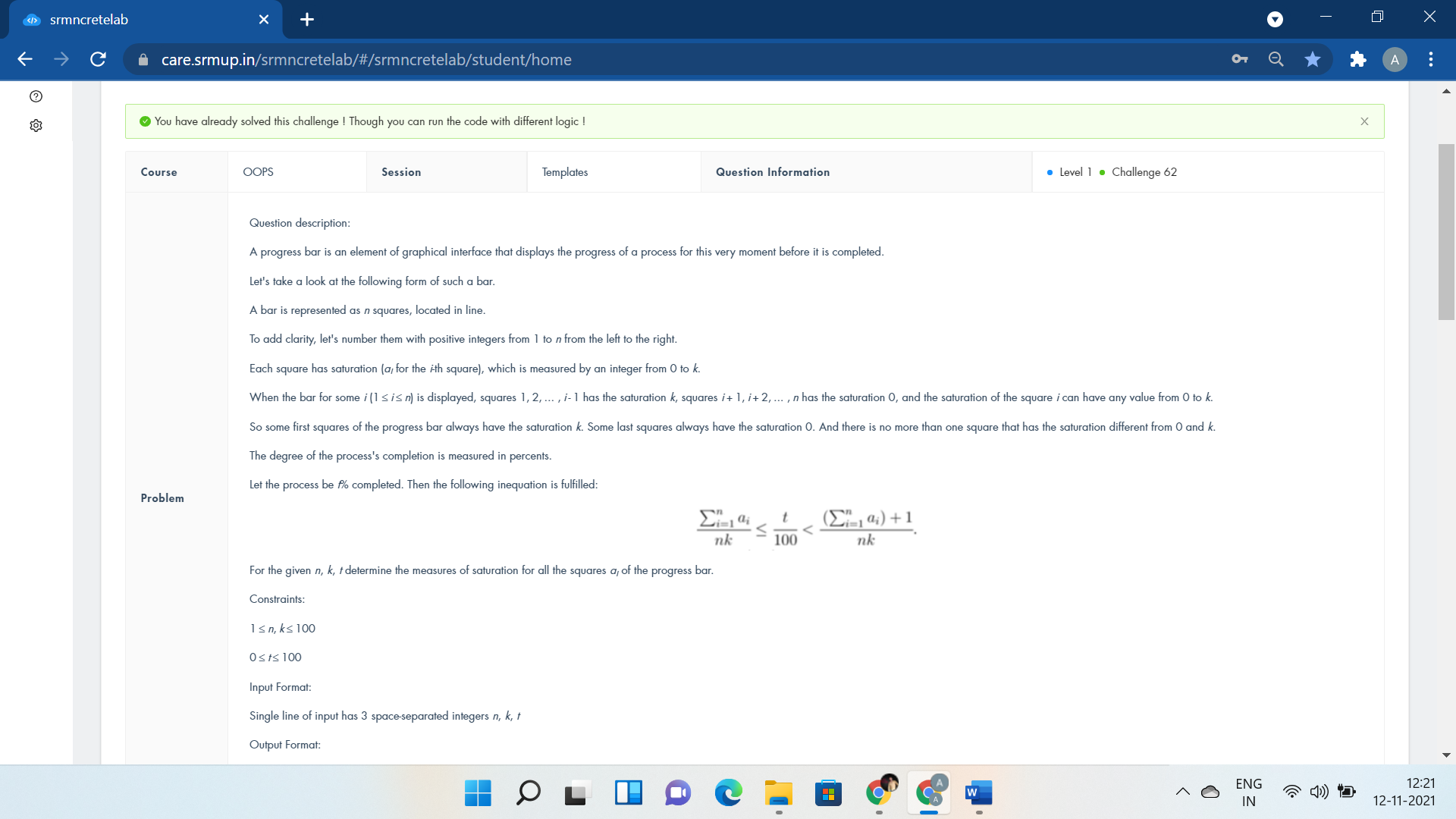
cout<<"Equal\n";

else

Visit(a,b);

return 0;

}



#include <iostream>

using namespace std;

template <class Interface>

Interface Bar(Interface n,Interface k,Interface t){

t = t\*k\*n/100.0;

while(n--){

cout<<min(t,k)<<" ";

t-=min(t,k);

}

return 1;

}

int main()

{

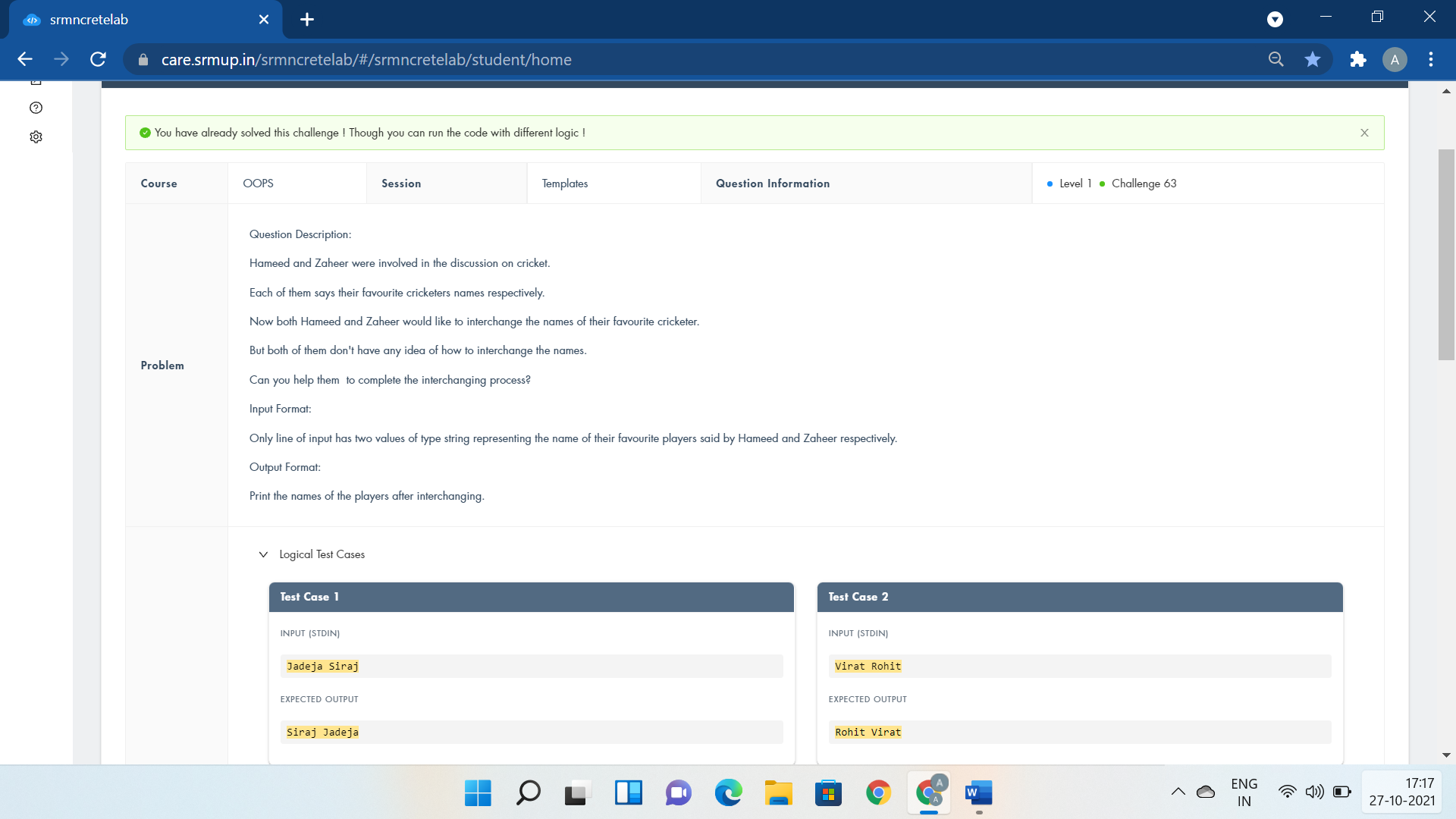
int n,k,t;

cin>>n>>k>>t;

Bar(n,k,t);

return 0;

}



#include <iostream>

using namespace std;

template <class T>

void InterchangeFavPlayers(T &player1,T &player2){

cout<<player2<<" "<<player1;

}

int main()

{

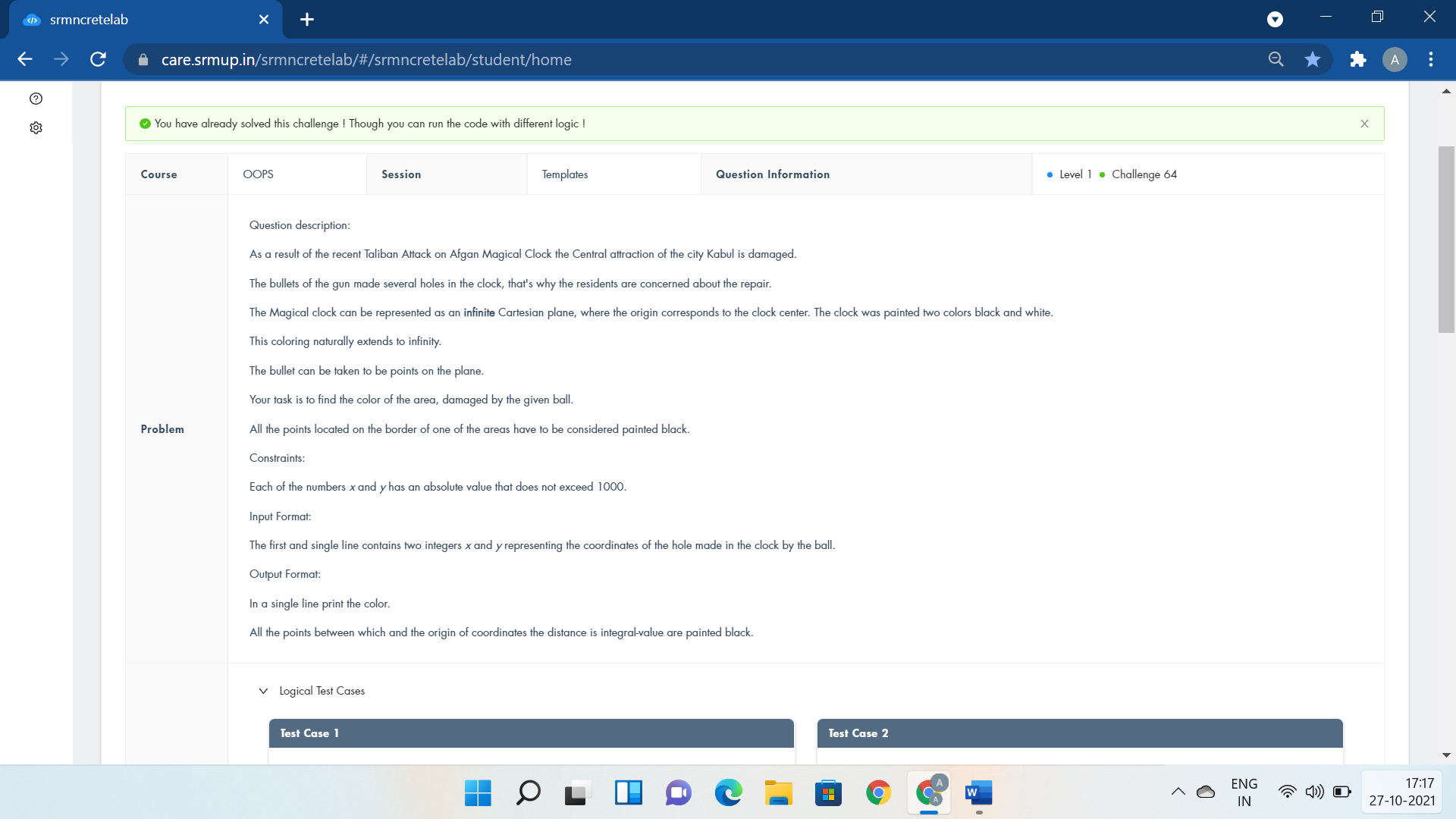
string player1,player2;

cin>>player1>>player2;

InterchangeFavPlayers(player1,player2);

return 0;

}



#include <iostream>

#include<cmath>

using namespace std;

template <class Hole>

Hole MagicClocl(Hole x,Hole y){

int c;

c=sqrt(x\*x+y\*y);

if(c\*c==x\*x+y\*y){

cout<<"black\n";

return 0;

}

if(x\*y<0)

c++;

if(c%2==0)

cout<<"black";

else cout<<"white";

return 1;

}

using namespace std;

int main()

{

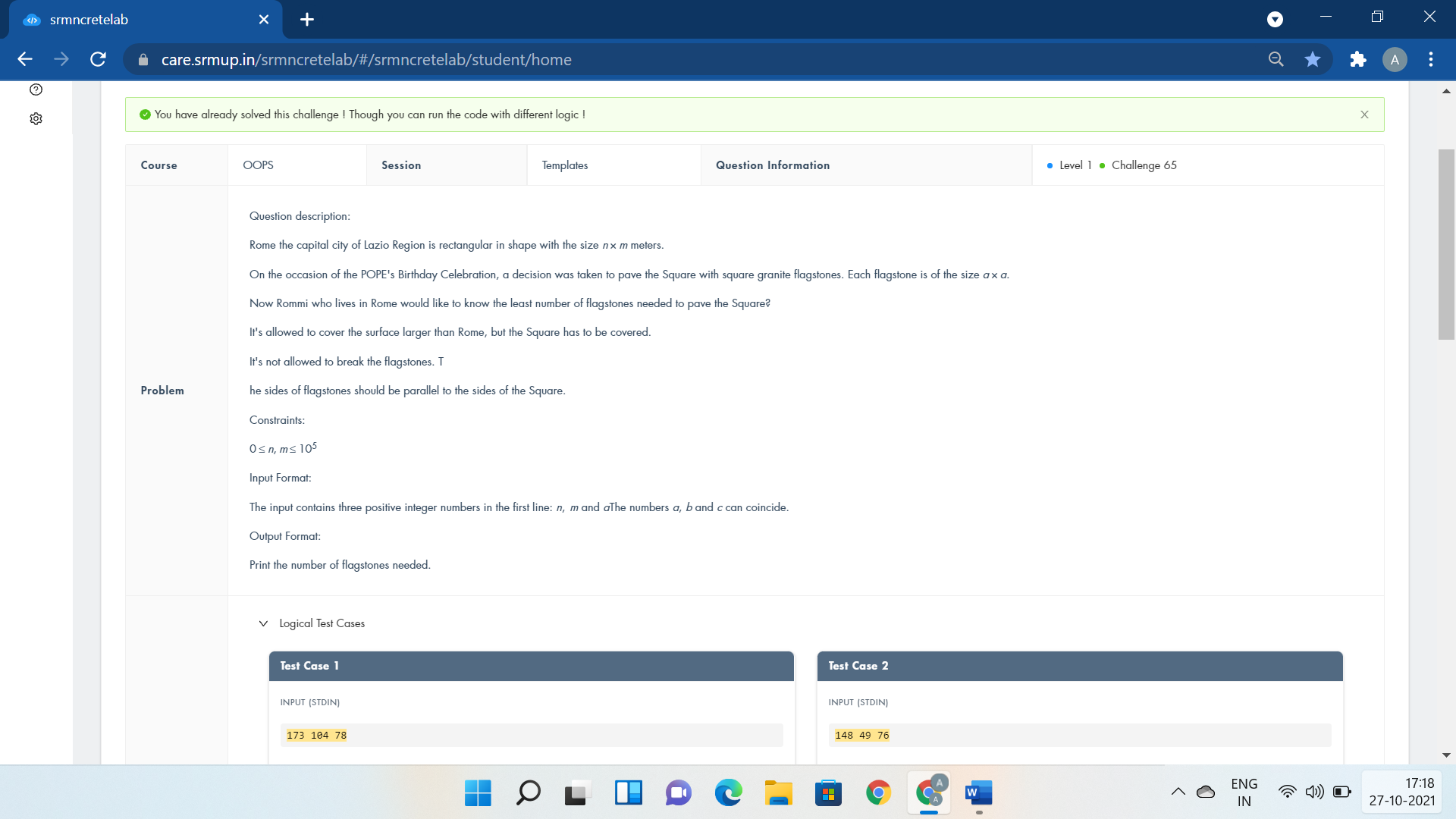
int x,y;

cin>>x>>y;

MagicClocl(x,y);

return 0;

}



#include <iostream>

using namespace std;

template <class Celebration>

Celebration Rome(Celebration a,Celebration b,Celebration c){

cout<<((b+c-1)/c)\*((a+c-1)/c);

return 1;

}

int main()

{

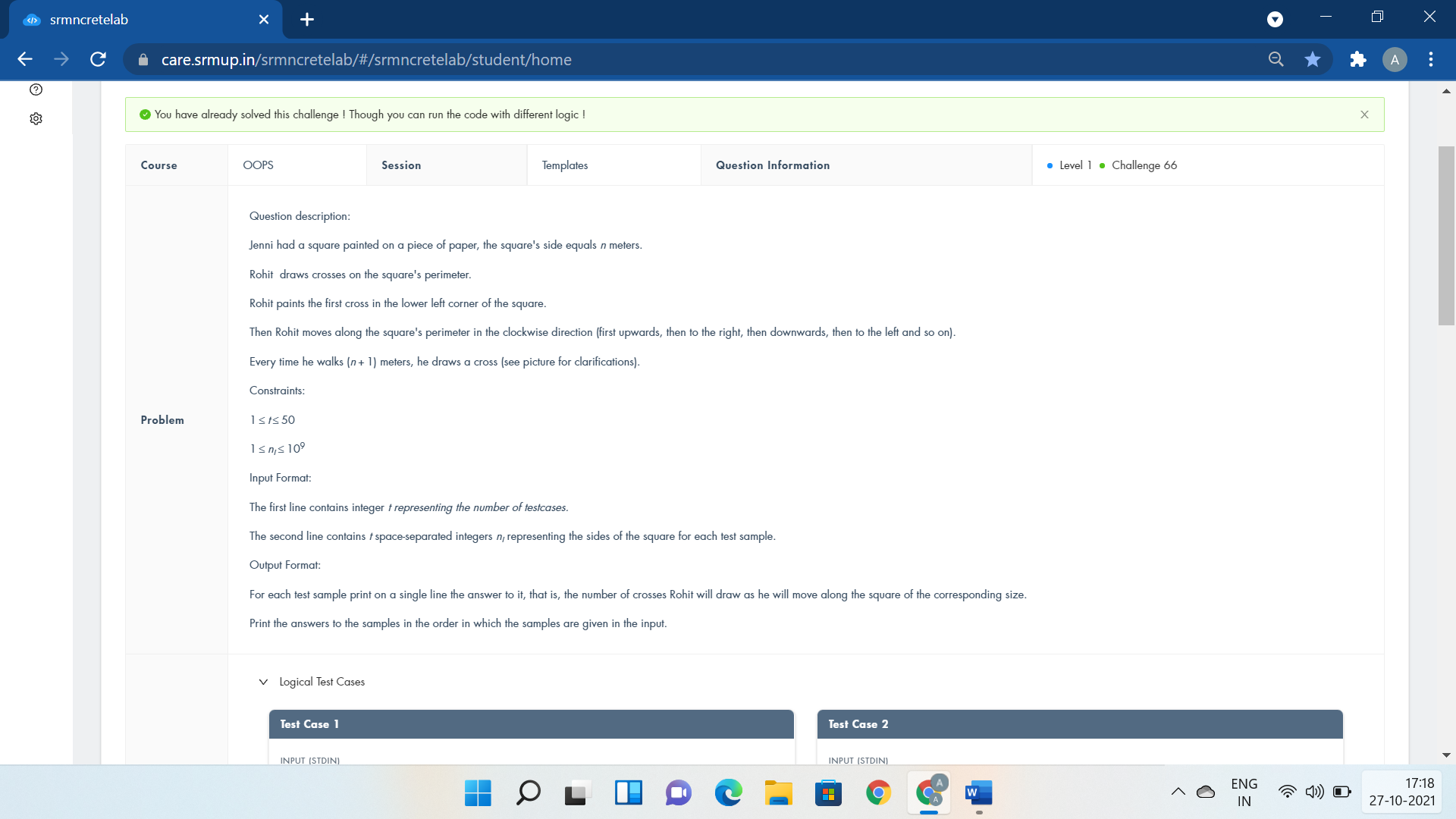
int a,b,c;

cin>>a>>b>>c;

Rome(a,b,c);

return 0;

}



#include <iostream>

using namespace std;

template <class Paper>

Paper Square(Paper T){

if(T%2==0)

return 4\*T+1;

else if(T%4==1)

return 2\*T+1;

else

return T+1;

}

int main()

{

int T,n;

cin>>T;

while(T--){

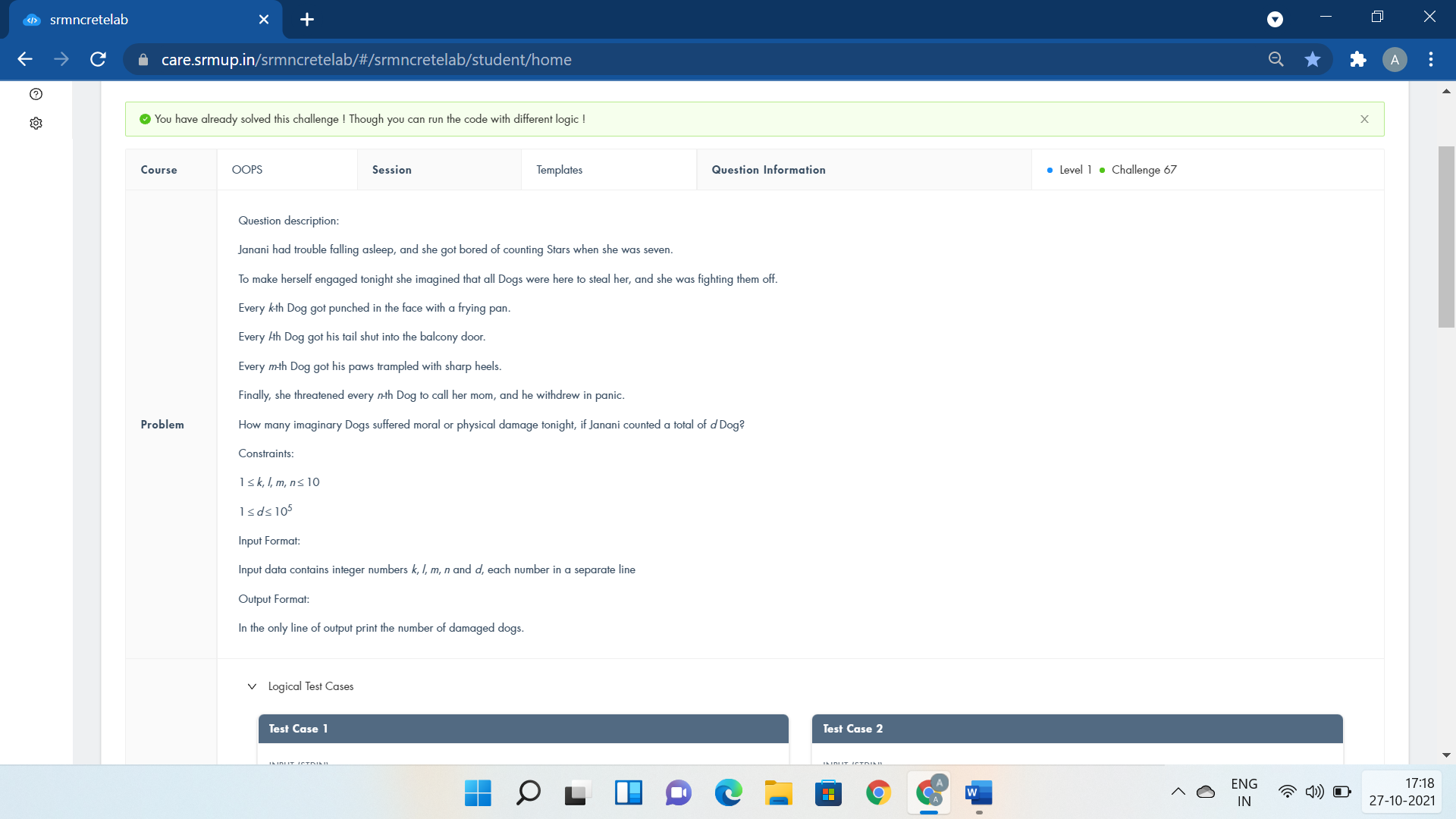
cin>>n;

cout<<Square(n)<<endl;

}

return 0;

}



#include <iostream>

using namespace std;

template <class LackofSleep>

LackofSleep Counting(LackofSleep k,LackofSleep l,LackofSleep m,LackofSleep n,LackofSleep d)

{

int c=0;

for(int i=0;i<=d;i++){

if(i%k==0||i%l==0||i%m==0||i%n==0)

c++;

}

return c-1;

}

int main()

{

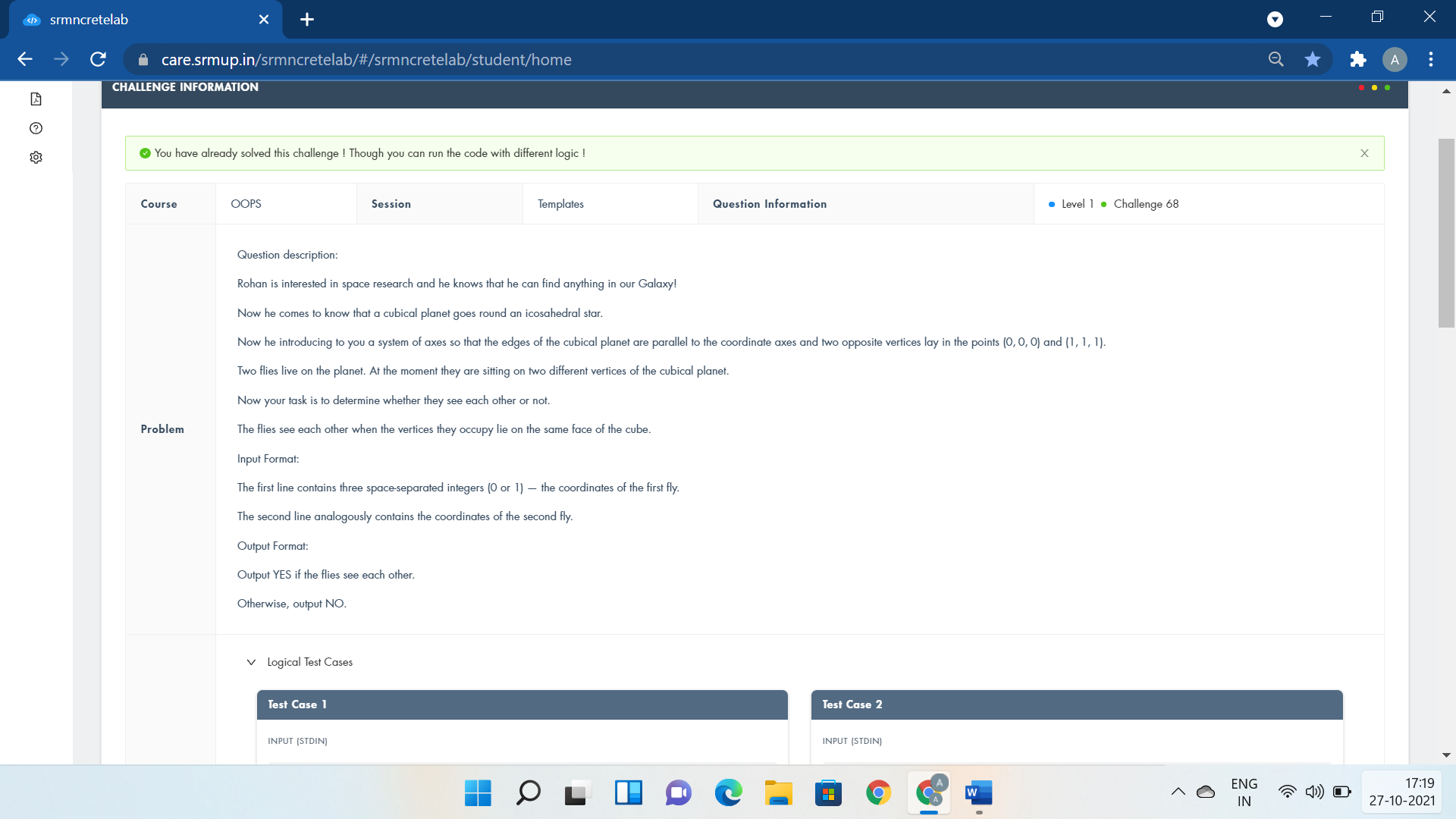
int k,l,m,n,d;

cin>>k>>l>>m>>n>>d;

cout<<Counting(k,l,m,n,d);

return 0;

}



#include <iostream>

using namespace std;

template <class Universe>

Universe Planet (Universe x1,Universe y1,Universe z1,Universe x2,Universe y2,Universe z2){

if(x1==x2 || y1 == y2 || z1==z2)

cout<<"YES";

else

cout<<"NO";

return 1;

}

int main()

{

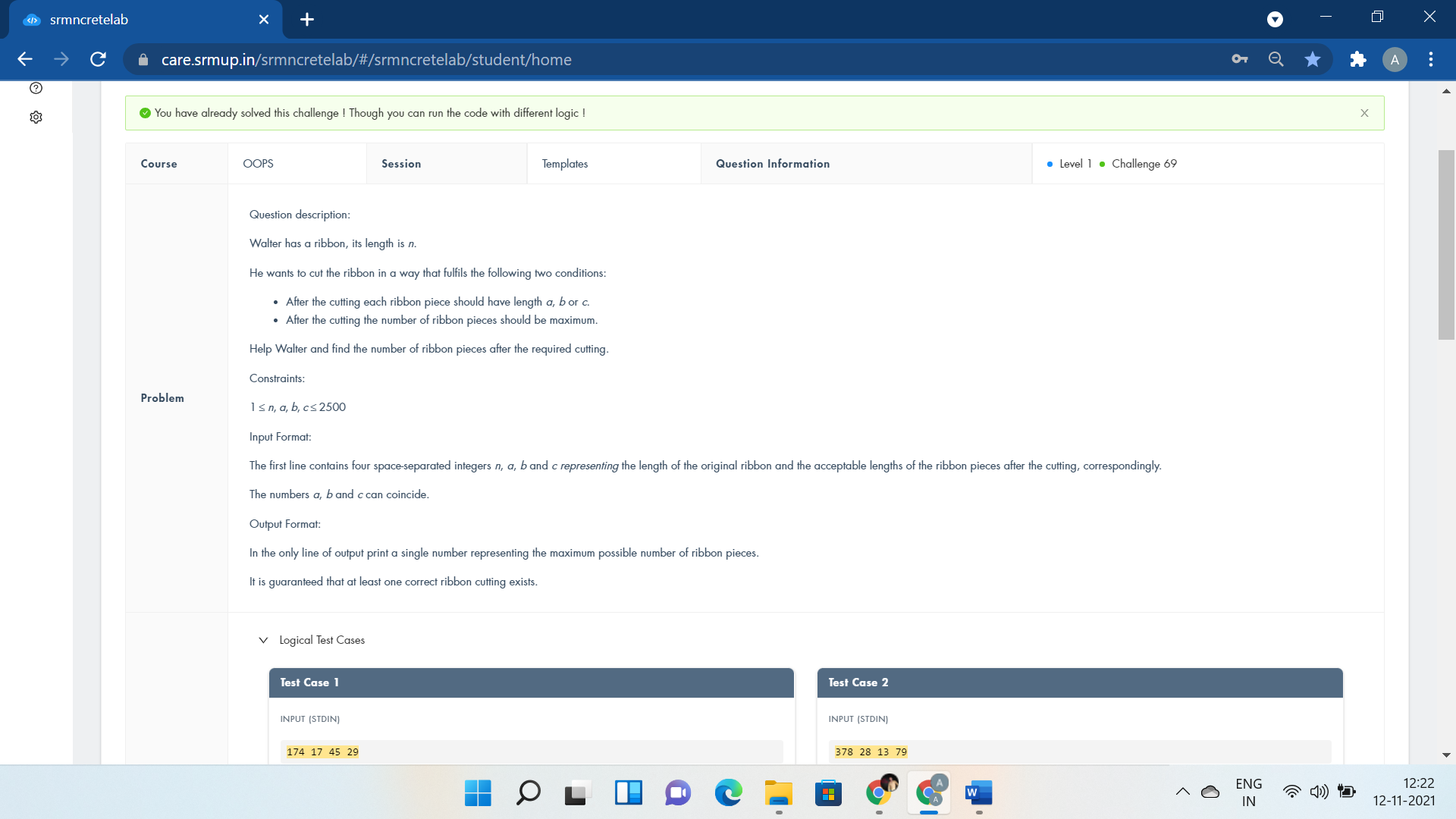
int x1,y1,z1,x2,y2,z2;

cin>>x1>>y1>>z1>>x2>>y2>>z2;

Planet(x1,y1,z1,x2,y2,z2);

return 0;

}



#include<bits/stdc++.h>

using namespace std;

template <class Ribbon>

Ribbon Pieces(Ribbon n,Ribbon a,Ribbon b,Ribbon c){

int d=1,e,i,j;

for(i=0;i<=4000;i++)

for(j=0;j<=4000;j++) {

e=n-a\*i-b\*j;

if(e>=0&&e%c==0)

d=max(d,i+j+e/c);

}

cout<<d;

return 1;

}

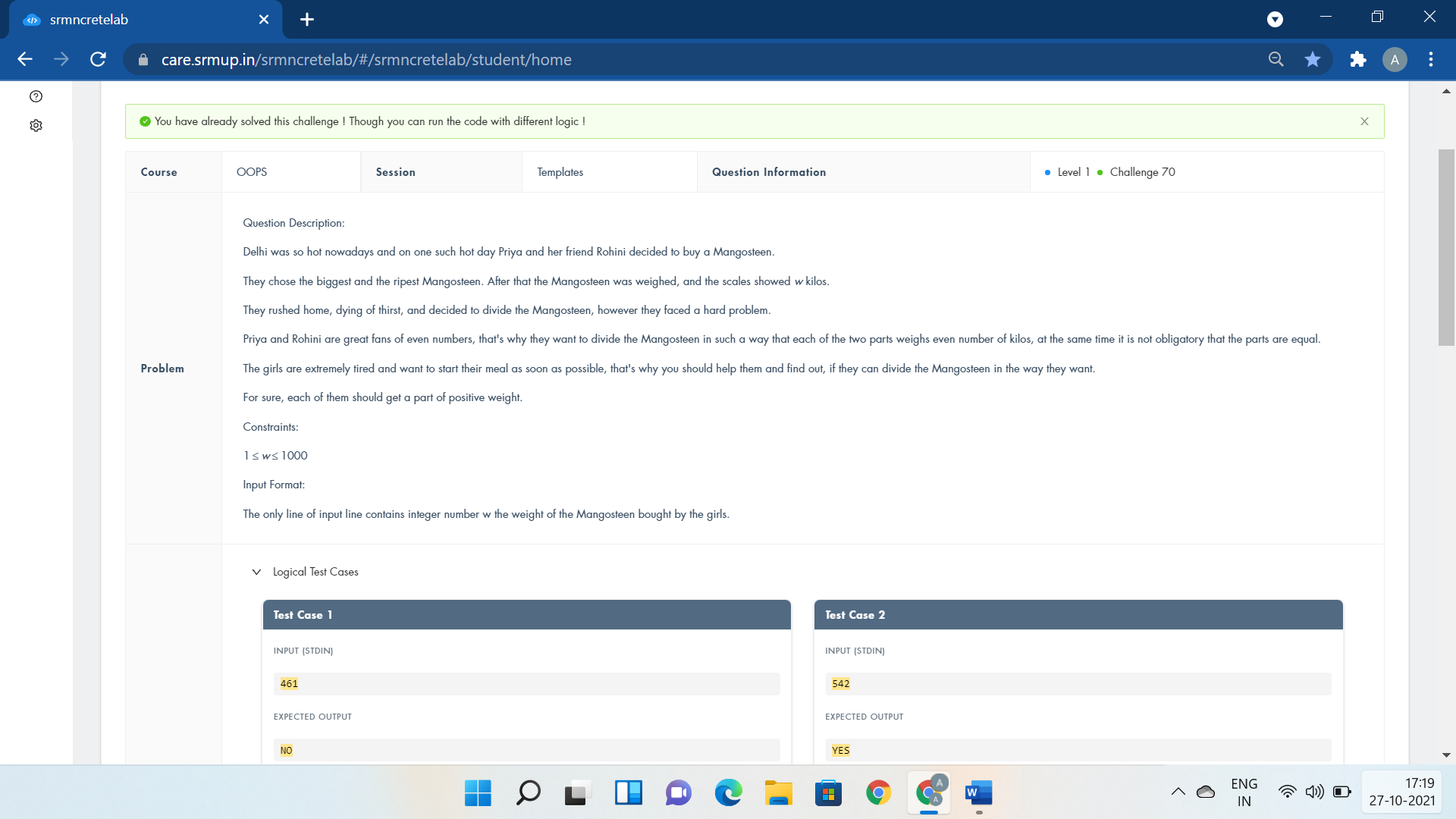
int main(){

int n,a,b,c;

cin>>n>>a>>b>>c;

Pieces(n,a,b,c);

}



#include <iostream>

using namespace std;

template<class T>

T DivideMangosteen(T PurchasedWeight){

if(PurchasedWeight%2==0)

cout<<"YES";

else

cout<<"NO";

return 1;

}

int main()

{

int PurchasedWeight;

cin>>PurchasedWeight;

DivideMangosteen(PurchasedWeight);

return 0;

}