NAME	**ANISH KUMAR**
UID	**20BCS1385**
SECTION	**DWWC-43**
DATE	**03-01-2023**

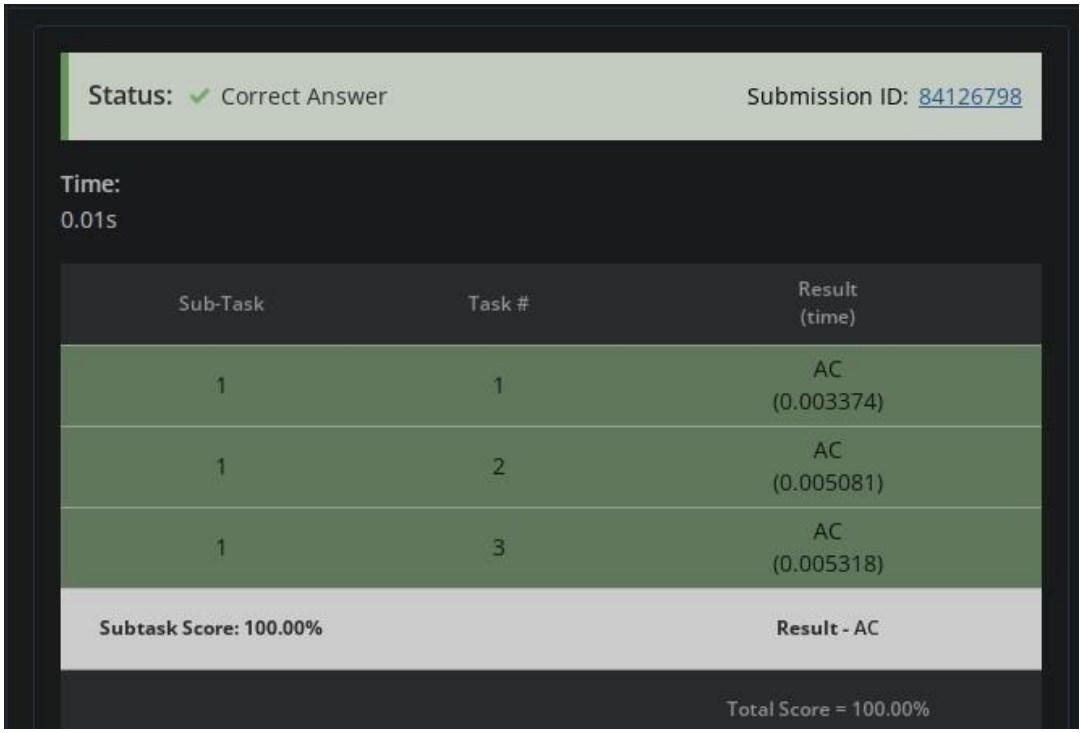
DAY 1 (Questions)

1. Gold Mining CODE:

```
#include <iostream> using

namespace std;



int main() {

        // your code goes

here   int t;   cin >> t;




    whi

le(t--){



        long long int n, x,

y;     cin >> n >> x >> y;

        long long int

friends = n+1;     long long

int capacity = friends * y;
```

```
if(x<=capacity){
cout << "YES"
<<endl;                }
else{           cout <<
"NO" << endl;       }
            }


             return 0;
}
```

OUTPUT:



2. Sum in a Triangle

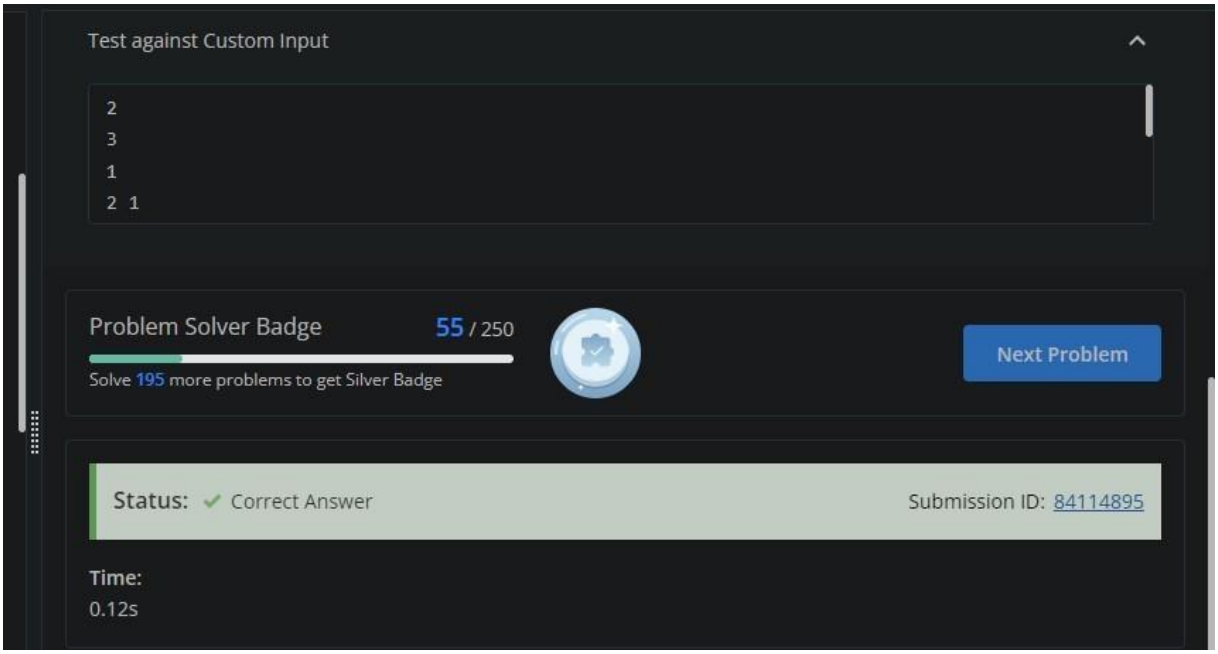
Code:

```
#include <bits/stdc++.h> using namespace std; #define ll unsigned long
long int ll f(int n,vector<vector<ll>> &a,int i,int j,vector<vector<ll>> &dp)
```

```
{    if(i==n-1)        return
a[n1][j];    if(dp[i][j]!=-1)
return dp[i][j];    ll
down=a[i][j]+f(n,a,i+1,j,dp);
ll
diag=a[i][j]+f(n,a,i+1,j+1,dp);
return
dp[i][j]=max(down,diag);
}

int main() {
   ll t;
   cin>>t;

   while(t--)
   {       int n;       cin>>n;
vector<vector<ll>> a(n,vector<ll> (n));       vector<vector<ll>>
dp(n,vector<ll> (n,-1));

      for(int i=0;i<n; i++){
for(int j=0;j<=i;j++)
         {
cin>>a[i][j];
         }

      }

cout<<f(n,a,0,0,dp)<<endl;
```

```
    }


    return 0;
}
```

OUTPUT:



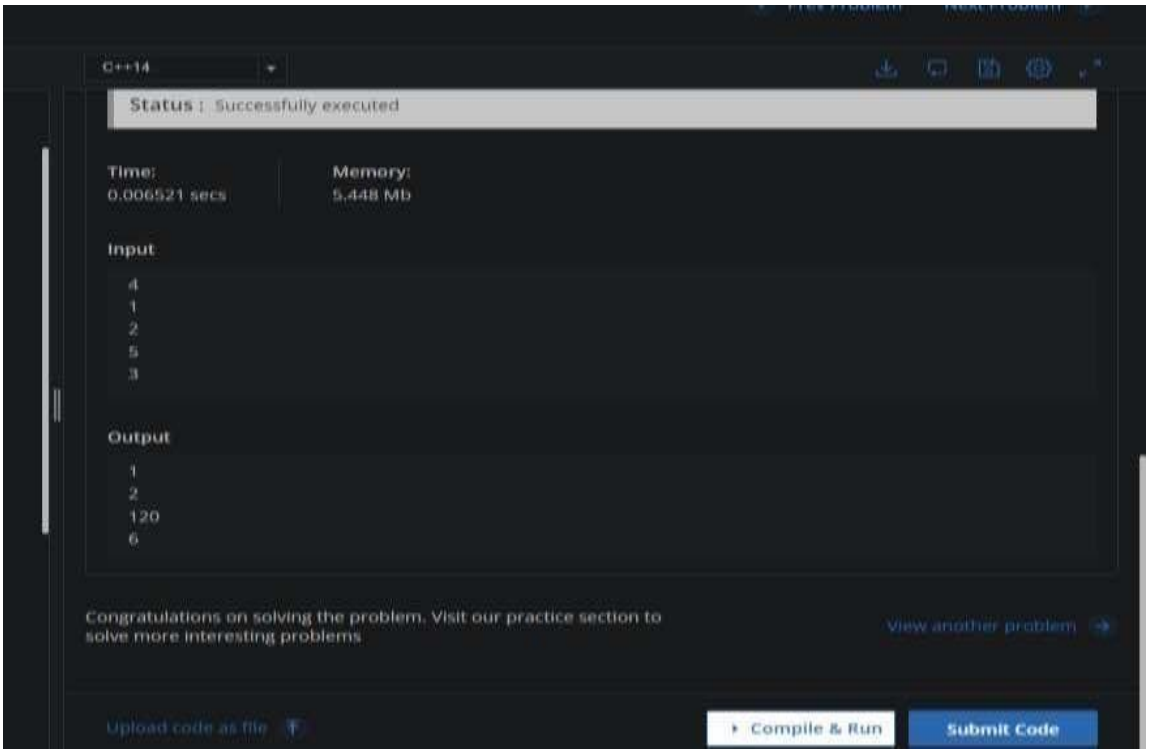
3. **Small factorial**

Code:

```
#include <bits/stdc++.h>

#include <iostream>

#include <boost/multiprecision/cpp_int.hpp>  using
namespace boost::multiprecision;  using namespace
std;


int main() {
            // your code goes here

 int tc ;            cin>>tc;
while(tc--)
```

```
        {
 int n;   cin>>n;    cpp_int
count = 1;    for(int i = 1;
i<=n; i++)
        {
                count = count *i;

        }
            cout <<count<<endl;
}
```

OUTPUT:

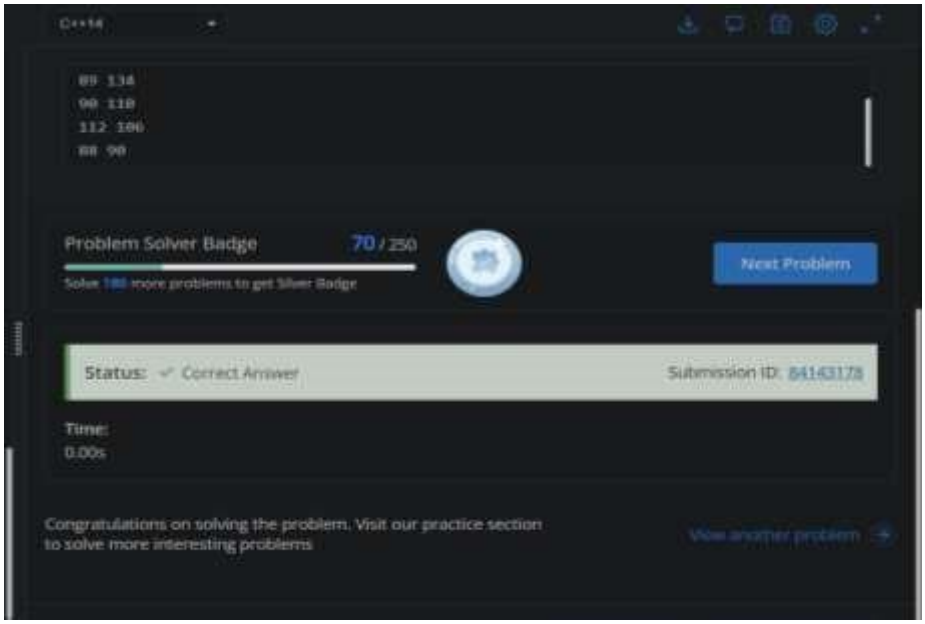


4. **Lead game**

Code:

```
#include<bits/stdc++.h>
using namespace std;


#define lli long long int

#define ll long long

#define no cout<<"NO \n";

#define yes cout<<"YES \n";
#define test lli trtyuio ;
cin>>trtyuio; while(trtyuio--)
int main(){
ios_base::sync_with_stdio(fa
lse);    cin.tie(NULL);    lli
n;cin>>n;    lli
a=0,b=0,ans,sum=0;    for
(int i = 0; i < n; i++)
   {
      int x,y;cin>>x>>y;
a+=x;b+=y;
if(abs(ab)>sum)
      {
if(a>b)
        {
            ans=1;sum=abs(a-
b);
        }
else
        {
            ans=2;sum=abs(a-
b);
        }
      }
   }
   cout<<ans<<" "<<sum;


   return 0;
}
```

OUTPUT:



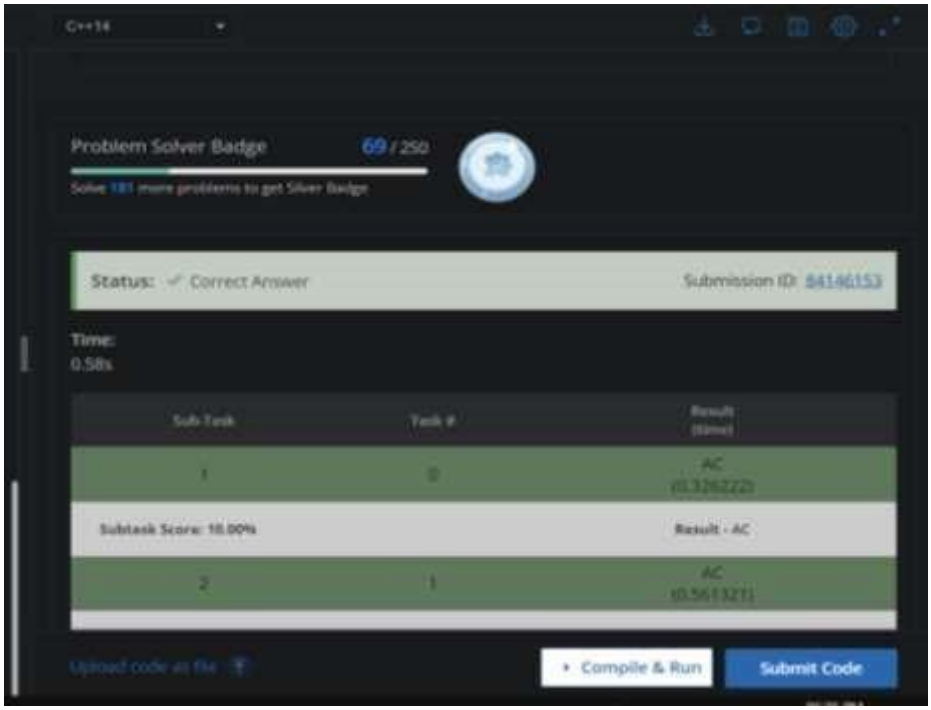
5. **Fire and Ice**

Code:

```
#include <stdio.h> #include<inttypes.h> void multiply(uint64_t
F[2][2], uint64_t M[2][2],uint64_t k); void power(uint64_t F[2][2],
uint64_t n,uint64_t k); uint64_t fib(uint64_t n,uint64_t k)

{

  uint64_t F[2][2] =
{{1,1},{1,0}};   if (n == 0)
return 0;   power(F, n-1,k);
return F[0][0];

}

void power(uint64_t F[2][2], uint64_t n,uint64_t k)

{

  if( n == 0 || n == 1)
return;   uint64_t M[2][2] =
{{1,1},{1,0}};


  power(F, n/2,k);
multiply(F, F,k);


  if (n%2 != 0)
multiply(F, M, k);

}
```

```
void multiply(uint64_t F[2][2], uint64_t M[2][2],uint64_t k)

{

  uint64_t x =  (F[0][0]*M[0][0] + F[0][1]*M[1][0])%k;   uint64_t y =
(F[0][0]*M[0][1] + F[0][1]*M[1][1])%k;   uint64_t z =
(F[1][0]*M[0][0] + F[1][1]*M[1][0])%k;   uint64_t w =
(F[1][0]*M[0][1] + F[1][1]*M[1][1])%k;


  F[0][0] = x;
F[0][1] = y;

  F[1][0] = z;

  F[1][1] = w;

}


/* Driver program to test above function */ int main()

{

  uint64_t n,k,t;
scanf("%llu",&t);   while(t--)

  {
            scanf("%llu",&n);
scanf("%llu",&k);
printf("%llu\n",
(2*fib(n,k))%k);
  }

  return 0;

}
```

OUTPUT:



Status: ✔ Correct Answer Submission ID: 84204694

Time:
0.00s

Sub-Task	Task #	Result (time)
1	1	AC (0.003749)
1	2	AC (0.003707)
1	3	AC (0.003972)
1	4	AC (0.003989)
Subtask Score: 100.00%		**Result - AC**

Total Score = 100.00%