

## WORKSHEET 1

**Student Name:** Jannat Baweja

**DOMAIN CAMP:** 03-01-2023 to 14-01-2023

**Subject Name:** IT Skills (DSA)

**UID:** 20BCS2050

**Section/Group:** DWWC-43

### Question 1. FIRE & ICE

```
Language: C++14

1  #include <stdio.h>
2
3  #include<inttypes.h>
4
5
6
7  void multiply(uint64_t F[2][2], uint64_t M[2][2],uint64_t k);
8
9
10
11 void power(uint64_t F[2][2], uint64_t n,uint64_t k);
12
13 uint64_t fib(uint64_t n,uint64_t k)
14
15 {
16
17     uint64_t F[2][2] = {{1,1},{1,0}};
18
19     if (n == 0)
20         return 0;
21
22     power(F, n-1,k);
23
24     return F[0][0];
25
26 }
27
28
29
30
31 /* Optimized version of power() in method 4 */
32
33 void power(uint64_t F[2][2], uint64_t n,uint64_t k)
34
```

```

34
35 {
36
37     if( n == 0 || n == 1)
38         return;
39
40     uint64_t M[2][2] = {{1,1},{1,0}};
41
42
43
44
45     power(F, n/2,k);
46
47     multiply(F, F,k);
48
49
50
51     if (n%2 != 0)
52         multiply(F, M, k);
53
54 }
55
56
57
58 void multiply(uint64_t F[2][2], uint64_t M[2][2],uint64_t k)
59 {
60
61 {
62
63     uint64_t x = (F[0][0]*M[0][0] + F[0][1]*M[1][0])%k;
64     uint64_t y = (F[0][0]*M[0][1] + F[0][1]*M[1][1])%k;
65     uint64_t z = (F[1][0]*M[0][0] + F[1][1]*M[1][0])%k;
66     uint64_t w = (F[1][0]*M[0][1] + F[1][1]*M[1][1])%k;
67
68
69
70
71
72     F[0][0] = x;
73     F[0][1] = y;
74     F[1][0] = z;
75     F[1][1] = w;
76
77
78
79
80
81 }
82

```

```

82
83 int main()
84 {
85 {
86
87     uint64_t n,k,t;
88
89     scanf("%llu",&t);
90
91     while(t--)
92     {
93
94
95         scanf("%llu",&n);
96
97         scanf("%llu",&k);
98
99         printf("%llu\n", (2*fib(n,k))%k);
100
101     }
102
103     return 0;
104
105 }

```

## SOLUTION:

Status: <span style="color: green;">✓</span> Correct Answer			Submission ID: <a href="#">84133067</a>
Score: 100	Time: 0.58s	Memory: 5.3M	
Sub-Task	Task #	Result (time)	
1	0	AC (0.323320)	
Subtask Score: 10.00%		Result - AC	
2	1	AC (0.559308)	
Subtask Score: 30.00%		Result - AC	
3	2	AC (0.579833)	
Subtask Score: 60.00%		Result - AC	
Total Score = 100.00%			

## Question 2. GOLD MINING

```

Language: C++14

1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     // your code goes here
6     int t;
7     cin>>t;
8     while(t--){
9         int N,X,Y;
10        cin>>N>>X>>Y;
11        int max_g = (N+1)*Y;
12
13        if(max_g>=X)
14            cout<<"YES"<<" ";
15        else
16            cout<<"NO"<<" ";
17    }
18    return 0;
19 }

```

## SOLUTION:

Status: <span style="color: green;">✓</span> Correct Answer			Submission ID: <a href="#">84135059</a>
Score: 100	Time: 0.01s	Memory: 5.3M	
Sub-Task	Task #	Result (time)	
1	1	AC (0.003820)	
1	2	AC (0.005502)	
1	3	AC (0.005810)	
Subtask Score: 100.00%		Result - AC	
Total Score = 100.00%			

### Question 3. SMALL FCATORIALS

```
Language: C++14

1 // We have populated the solutions for the 10 easiest problems for your support.
2 // Click on the SUBMIT button to make a submission to this problem.
3
4 #include <bits/stdc++.h>
5 #include <boost/multiprecision/cpp_int.hpp>
6 #include <iostream>
7 using namespace std;
8 using namespace boost::multiprecision;
9
10 cpp_int fact(int n){
11
12     if (n<=1){
13         return 1;
14     }
15     return n*fact(n-1);
16 }
17
18 int main() {
19     // your code goes here
20     int t;
21     cin>>t;
22     while(t-->0)
23     {
24         int n;
25         cin>>n;
26         cout<<fact(n)<<endl;
27     }
28
29     return 0;
30 }
31
```

### SOLUTION:

Status: ✓ Correct Answer Submission ID: [84132016](#)


Time:	Memory:
0.00s	5.3M

### Question 4. SUM OF DIGITS

```
Language: C++14

1 // We have populated the solutions for the 10 easiest problems for your support.
2 // Click on the SUBMIT button to make a submission to this problem.
3
4 #include <iostream>
5 using namespace std;
6
7 int main()
8 {
9     int t;
10    cin>>t;
11
12    while(t-->0)
13    {
14        int n;
15        cin>>n;
16
17        int sum=0;
18
19        while(n>0)
20        {
21            sum+=(n%10);
22            n/=10;
23        }
24
25        cout<<sum<<"\n";
26    }
27
28    return 0;
29 }
30
```

**SOLUTION:**

Status:  Correct Answer Submission ID: [84132934](#)


Time:  
0.00s

Memory:  
5.1M

**Question 5. THE LEAD GAME**

```
Language: C++14
1  #include <bits/stdc++.h>
2  using namespace std;
3
4  int main() {
5      int t;
6      cin>>t;
7      int max=0,leadp=0,c1=0,c2=0;
8      for(int i=0;i<t;i++){
9          int x,y;
10         cin>>x>>y;
11         c1+=x;
12         c2+=y;
13         if(c1>c2){
14             int lead=c1-c2;
15             if(lead>max){
16                 max=lead;
17                 leadp=1;
18             }
19         }
20         else{
21             int lead=c2-c1;
22             if(lead>max){
23                 max=lead;
24                 leadp=2;
25             }
26         }
27     }cout<<leadp<<" "<<max;
28     // your code goes here
29     return 0;
30 }
```

**SOLUTION:**

Status:  Correct Answer Submission ID: [84198770](#)

Time:  
0.01s

Memory:  
5.3M

## Question 6. SUMS IN A TRIANGLE

```
Language: C++14

1  #include <bits/stdc++.h>
2  using namespace std;
3
4  int solve(int i,int j,vector<vector<int>> tri,int n,vector<vector<int>> &dp)
5  {
6      if(i==n-1)
7          return tri[i][j];
8      if(dp[i][j]!=-1)
9          return dp[i][j];
10     int below=tri[i][j] + solve(i+1,j,tri,n,dp);
11     int diag=tri[i][j] + solve(i+1,j+1,tri,n,dp);
12     return dp[i][j]=max(below,diag);
13 }
14
15 int main() {
16     // your code goes here
17     int t;
18     cin>>t;
19     while(t-->0)
20     {
21         int n;
22         cin>>n;
23         vector<vector<int>> tri;
24         for(int i=1;i<=n;i++)
25         {
26             vector<int> temp;
27             for(int j=1;j<=i;j++)
28             {
29                 int x;
30                 cin>>x;
31                 temp.push_back(x);
32             }
33             tri.push_back(temp);
34         }
35     }
36 }
```

```

35     }
36     vector<vector<int>> dp(n,vector<int>(n,0));
37
38     for(int k=0;k<n;k++)
39         dp[n-1][k]=tri[n-1][k];
40
41     for(int i=n-2;i>=0;i--)
42     {
43         for(int j=i;j>=0;j--)
44         {
45             int below=tri[i][j]+dp[i+1][j];
46             int diag=tri[i][j]+dp[i+1][j+1];
47             dp[i][j]=max(below,diag);
48         }
49     }
50     cout<<dp[0][0]<<endl;
51 }
52
53 return 0;
54 }
```

## SOLUTION:

Status: ✔ Correct Answer

Submission ID: [84199142](#)

Time: 0.12s

Memory: 5.3M

## Question 7. CHEF ON VACATION

```

Language: C++14
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     // your code goes here
6     int t;
7     cin >> t;
8
9     while(t--){
10         long long int x,y,z;
11         cin >>x>>y>>z;
12
13         long long int goVacations = x+y;
14         if(goVacations <= z){
15             cout<<"YES"<<endl;
16         }
17         else{
18             cout<<"NO"<<endl;
19         }
20     }
21     return 0;
22 }
0:0

```

## SOLUTION:

Status: ✔ Correct Answer

Submission ID: [84133972](#)

Score:  
100

Time:  
0.01s

Memory:  
5.3M

Sub-Task	Task #	Result (time)
1	0	AC (0.004150)
1	1	AC (0.006497)
Subtask Score: 100.00%		Result - AC
Total Score = 100.00%		