

WORKSHEET 1

NAME- EKROOP SINGH

SEC-DWWC 43

UID:21BCS8143

Date- 03/01/2023

1. Fire and Ice

Program Code:

```

1  #include <stdio.h>
2  #include <stdint.h>
3
4  void multiply(uint64_t F[2][2], uint64_t M[2][2], uint64_t k);
5
6  void power(uint64_t F[2][2], uint64_t n, uint64_t k);
7
8  /* Function that returns nth fibonacci number */
9  uint64_t Fib(uint64_t n, uint64_t k)
10 {
11     uint64_t F[2][2] = {{1,1},{1,0}};
12     if (n == 0)
13         return 0;
14     power(F, n, k);
15     return F[0][0];
16 }
17
18 /* Optimized version of power(). In method a */
19 void power(uint64_t F[2][2], uint64_t n, uint64_t k)
20 {
21     if (n == 0 || n == 1)
22         return;
23     uint64_t M[2][2] = {{1,1},{1,0}};
24
25     power(F, n/2, k);
26     multiply(F, M, k);
27
28     if (n%2 != 0)
29         multiply(F, M, k);
30 }
31
32 void multiply(uint64_t F[2][2], uint64_t M[2][2], uint64_t k)
33 {
34     uint64_t x = (F[0][0]*M[0][0] + F[0][1]*M[1][0])%k;
35     uint64_t y = (F[0][0]*M[0][1] + F[0][1]*M[1][1])%k;
36     uint64_t z = (F[1][0]*M[0][0] + F[1][1]*M[1][0])%k;
37     uint64_t w = (F[1][0]*M[0][1] + F[1][1]*M[1][1])%k;
38
39     F[0][0] = x;
40     F[0][1] = y;
41     F[1][0] = z;
42     F[1][1] = w;
43 }
44
45 int main()
46 {
47     uint64_t n, k;
48
49     scanf("%llu", &n);
50     while(n--)
51     {
52         scanf("%llu", &k);
53         printf("%llu", (2*Fib(n, k))%k);
54     }
55     return 0;
56 }

```

```

38
39 if (n%2 != 0)
40     multiply(F, M, k);
41 }
42
43 void multiply(uint64_t F[2][2], uint64_t M[2][2], uint64_t k)
44 {
45     uint64_t x = (F[0][0]*M[0][0] + F[0][1]*M[1][0])%k;
46     uint64_t y = (F[0][0]*M[0][1] + F[0][1]*M[1][1])%k;
47     uint64_t z = (F[1][0]*M[0][0] + F[1][1]*M[1][0])%k;
48     uint64_t w = (F[1][0]*M[0][1] + F[1][1]*M[1][1])%k;
49
50     F[0][0] = x;
51     F[0][1] = y;
52     F[1][0] = z;
53     F[1][1] = w;
54 }
55
56 int main()
57 {
58     uint64_t n, k;
59
60     scanf("%llu", &n);
61     while(n--)
62     {
63         scanf("%llu", &k);
64         printf("%llu", (2*Fib(n, k))%k);
65     }
66     return 0;
67 }

```

Test against Custom Input

Output:

Subtask Info		
Status: ✓ Correct Answer		Submission ID: 84133293
Score: 100	Time: 0.58s	Memory: 5.2M
Sub-Task	Task #	Result (time)
1	0	AC (0.323468)
Subtask Score: 10.00%		Result - AC
2	1	AC (0.559455)
Subtask Score: 30.00%		Result - AC
3	2	AC (0.578799)
Subtask Score: 60.00%		Result - AC
Total Score = 100.00%		

2. Gold Mining

Program Code:

```
C++17
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int t;
6     cin>>t;
7     while(t--){
8         int n,x,y;
9         cin>>n>>x>>y;
10
11         double b = y*(n+1);
12         if(x <= b ){
13             cout<<"yes"<<endl;
14         }
15         else
16             cout<<"no"<<endl;
17     }
18     return 0;
19 }
20
```

Output

Output

```
no
yes
yes
```

3. The Lead Game

Program Code:

```
int lead=0;
int winner=0;
int player1=0,player2=0;

while(n--){
    int p1,p2;
    cin>>p1>>p2;

    player1+=p1;
    player2+=p2;

    int d = abs(player2-player1);

    if(d>lead){
        lead = d;
        player1>player2?winner=1:winner=2;
    }
}

cout<<winner<<" "<<lead<<endl;
```

Output:

Output

1 58

4. Sums in a triangle

Program Code:

```
int t;
cin>>t;
while(t--){
    int n;
    cin>>n;

    int a[n][n];

    for(int i=0;i<n;i++){
        for(int j=0;j<i+1;j++){
            cin>>a[i][j];
        }
    }

    for(int i=n-2;i>=0;i--){
        for(int j=0;j<i+1;j++){
            a[i][j]=a[i][j]+max(a[i+1][j],a[i+1][j+1]);
        }
    }
    cout<<a[0][0]<<endl;
```

16:3

Output:

Output

5

9

5.Small Factorials

Program Code:

```
int main() {  
    // your code goes here  
    int t;  
    cin>>t;  
    while(t--)  
    {  
        int n;  
        cin>>n;  
        cpp_int fact=1;  
        for(int i=n;i>0;i--)  
            fact=fact*i;  
        cout<<fact<<endl;  
    }  
    return 0;  
}
```

0:0

Output

Output

1

2

120

6