

## Problem 01

You are given an positive integer N (Here N means number of terms) . Now print the following sequence as shown below.

For N=10 , the sequence should be

0 1 1 2 3 5 8 13 21 34

**Sample Input 1:**

10

**Sample Input 2:**

1

**Sample Input 3:**

2

**Sample Output 1:**

0 1 1 2 3 5 8 13 21 34

**Sample Output 2:**

0

**Sample Output 3:**

0 1

Hints – Basically It's a Fibonacci sequence. The Fibonacci sequence is a sequence where the next term is the sum of the previous two terms. The first two terms of the Fibonacci sequence are 0 followed by 1.

>> Implement it using array.

```
1 #include <stdio.h>
2 int main()
3 {
4     int N, i;
5     scanf("%d", &N);
6     int array[N];
7     array[1]=0;
8     array[2]=1;
9     for (i=1; i<=N; i++)
10     {
11         array[i+2]=array[i]+array[i+1];
12     }
13     for (i=1; i<=N; i++)
14     {
15         printf("%d ", array[i]);
16     }
17     return 0;
18 }
19
```

## Problem 02

Write a C program to take one positive integer **N**, the size of an array as input. Then take a positive integer array of size **N** as input and sort the array in ascending order. It is guaranteed that the input array will contain distinct integers.

Note – It is possible to sort an array without any sorting algorithm. Hence for this problem you can't use any sorting algorithm.

**Sample Input 1:**

5

6 1 5 3 17

**Sample Input 2:**

7

5 2 15 7 9 1 10

**Sample Output 1:**

1 3 5 6 17

**Sample Output 2:**

1 2 5 7 9 10 15

Hints – Check the 31<sup>st</sup> October conceptual session.

```
1 #include <stdio.h>
2 int main()
3 {
4     int N, i, j, temp;
5     scanf("%d", &N);
6     int array[N];
7     for (i=1; i<=N; i++)
8     {
9         scanf("%d", &array[i]);
10    }
11    for (i=1; i<=N; i++)
12    {
13        for (j=i+1; j<=N; j++)
14        {
15            if(array[i]>array[j])
16            {
17                temp= array[i];
18                array[i]= array[j];
19                array[j]= temp;
20            }
21        }
22    }
23    for (i=1; i<=N; i++)
24    {
25        printf("%d ", array[i]);
26    }
27    return 0;
28 }
```

## Problem 03

Write a C program to take one positive integer **N**, the size of an array as input. Then take a positive integer array of size **N** as input and tell if the sum of odd values is even or not.

If the sum of odd values is even print YES otherwise NO.

**Sample Input 1:**

5

5 2 7 4 6

**Sample Output 1:**

YES

**Sample Input 2:**

5

3 17 8 10 3

**Sample Output 2:**

NO

```
1  #include <stdio.h>
2  int main()
3  {
4      int N, i, sum=0;
5      scanf("%d", &N);
6      int array[N];
7      for (i=1; i<=N; i++)
8      {
9          scanf("%d", &array[i]);
10     }
11     for (i=1; i<=N; i++)
12     {
13         if(array[i]%2==1)
14             sum = sum+array[i];
15     }
16     if (sum%2==1)
17         printf("NO");
18     else
19         printf("YES");
20     return 0;
21 }
```

## Problem 04

Write a C program to take one positive integer **N**, the size of an array as input. Then take a positive integer array of size **N** . And next line will contain **k** . Now find the **k**-th smallest element from the array.

**Sample Input:**

6  
6 25 1 4 9 17  
3

**Sample Output:**

6

Explanation – For the above test case  $k=3$  means you need to find the 3<sup>rd</sup> smallest element from the array

```
1  #include <stdio.h>
2  int main()
3  {
4      int N, i, j, temp, k;
5      scanf("%d", &N);
6      int array[N];
7      for (i=1; i<=N; i++)
8      {
9          scanf("%d", &array[i]);
10     }
11     scanf("%d", &k);
12     for (i=1; i<=N; i++)
13     {
14         for (j=i+1; j<=N; j++)
15         {
16             if(array[i]>array[j])
17             {
18                 temp=array[i];
19                 array[i]=array[j];
20                 array[j]=temp;
21             }
22         }
23     }
24     for (i=1; i<=N; i++)
25     {
26         if(i==k)
27             printf("%d", array[i]);
28     }
29     return 0;
30 }
```

## Problem 05

Write a C program to take one positive integer **N**, the size of an array as input. Then take a positive integer array of size **N** . Now count the number of prime numbers from this array and print them.

In the sample output the first line will contain count of prime numbers and second line will contain the prime numbers.

### Sample Input 1:

5  
2 10 4 21 97

### Sample Output 1:

2  
2 97

### Sample Input 2:

6  
5 12 16 19 23 9

### Sample Output 2:

3  
5 19 23

Hints – Nested loop

```
1  #include <stdio.h>
2
3  int main()
4  {
5      int N, i, j, flag, t, count=0;
6      scanf("%d", &N);
7      int arr[N], prime[100];
8      for (i = 0; i < N; i++)
9      {
10         scanf("%d", &arr[i]);
11     }
12     for (i = 0; i < N; i++)
13     {
14         flag = 0;
15         for (j = 2; j <= arr[i]/2; j++)
16         {
17             if(arr[i]%j == 0)
18             {
19                 flag = 1;
20                 break;
21             }
22         }
23         if(flag == 0)
24         {
25             prime[t]=arr[i];
26             t++;
27             count++;
28         }
29     }
30     printf("%d\n", count);
31     for(i=0; i<t; i++)
32     {
33         printf("%d ", prime[i]);
34     }
35     return 0;
36 }
```

## Problem 06

Write a C program to take a positive integer **N** and print the pattern as shown below

For N=4, the pattern should be

```
1 2 3 4
 1 2 3
  1 2
   1
```

Sample Input:

Sample Output:

```
1 2 3 4 5 6
 1 2 3 4 5
  1 2 3 4
   1 2 3
    1 2
     1
```

6

```
1 #include <stdio.h>
2 int main()
3 {
4     int N, i, j;
5     scanf("%d", &N);
6     for(i=N; i>=1; i--)
7     {
8         for(j=1; j<=i; j++)
9         {
10             printf("%d ", j);
11         }
12         printf("\n");
13         for(j=N-1; j>=i-1; j--)
14         {
15             printf(" ");
16         }
17     }
18     return 0;
19 }
```



### Codeforces Problem Link –

- 1) <https://codeforces.com/contest/707/problem/A>
- 2) <https://codeforces.com/contest/1472/problem/A>
- 3) <https://codeforces.com/problemset/problem/1676/B>

```
1 // Brain's Photos: Codeforces
2
3 #include<stdio.h>
4 int main()
5 {
6     int a, b, flag=0, i;
7     char color;
8
9     scanf("%d%d", &a, &b);
10
11     for(i=0; i<a*b*2; i++)
12     {
13         scanf("%c", &color);
14         if(color=='C' || color=='M' || color=='Y')
15             flag =1;
16     }
17
18     if(flag)
19         printf("#Color");
20
21     else
22         printf("#Black&White");
23
24     return 0;
25 }
26
```

// Equal Candies: Codeforces

```
#include<stdio.h>
int main()
{
    int t;
    scanf("%d", &t);

    for(int i=0; i<t; i++)
    {
        int n;
        scanf("%d", &n);
        int a[n];
        for(int j=0; j<n; j++)
        {
            scanf("%d", &a[j]);
        }
        int sum=0;
        int min=a[0];
        for(int j=0; j<n; j++)
        {
            sum=sum+a[j];
            if(a[j]<min)
            {
                min=a[j];
            }
        }
        printf("%d\n", sum-(n*min));
    }
    return 0;
}
```

// Cards for Friends: Codeforces

```
#include<stdio.h>
int main()
{
    int t, i, w, h;
    long long int n, sheet;
    scanf("%d", &t);

    int result[t];
    for(i=1; i<=t; i++)
    {
        scanf("%d %d %lld", &w, &h, &n);
        sheet=1;
        while(w%2==0 || h%2==0)
        {
            if(w%2==0)
            {
                w=w/2;
                sheet=sheet*2;
            }
            if(h%2==0)
            {
                h=h/2;
                sheet*=2;
            }
        }
        if(sheet>=n)
            result[i]=1;
        else
            result[i]=0;
    }
    for(i=1; i<=t; i++)
    {
        if(result[i]==1)
            printf("YES\n");
        else
            printf("NO\n");
    }
}
```