Started on	Friday, 29 December 2023, 2:07 PM
State	Finished
Completed on	Friday, 29 December 2023, 4:05 PM
Time taken	1 hour 57 mins

Question **1**

Correct

Marked out of 25.00

Dr. CooCoo is demanding some weird arrays. This is what she wants:

Given a number n, she wants an array of size n*n with the elements as follows:

n	Array
2	0 **1 2 1**
3	0 0 **1** 0 **2 1 3 2 1**
4	0 0 0 **1** 0 0 **2 1** 0 **3 2 1 4 3 2 1**

And so on...

Note: n is greater than or equal to 2

Write a program that takes input an integer n and returns the required array.

Example Input: 2

Output: 0 1 2 1

Example Input: 4

Output: 0 0 0 1 0 0 0 2 1 0 3 2 1 4 3 2 1

For example:

Input	Result
2	0 1 2 1
4	0 0 0 1 0 0 2 1 0 3 2 1 4 3 2 1

Answer: (penalty regime: 0 %)

```
import java.util.*;
public class Array1
{
    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
        int a = 0, size=n*n;
        int[] arr = new int[size];

        for(int i=1;i<=n;i++)
        {
            for(int k=0;k<n-i;k++)
            {
                arr[a]=0;
                a+=1;
            }
            for(int j=i;j>=1;j--)
            {
                arr[a]=j;
                a+=1;
            }
        }
        for(int i=0;i<size;i++)
        {
            System.out.print(arr[i]+" ");
        }

    }
}
```

	Input	Expected	Got	
✔	2	0 1 2 1	0 1 2 1	✔
✔	4	0 0 0 1 0 0 2 1 0 3 2 1 4 3 2 1	0 0 0 1 0 0 2 1 0 3 2 1 4 3 2 1	✔

Passed all tests! ✔

Question **2**

Correct

Marked out of 25.00

Dr. CooCoo has demanded another weird array!

Given a number n, she wants an array of the following pattern:

n	Array
2	1 1 2
3	1 1 2 1 2 3
4	1 1 2 1 2 3 1 2 3 4

And so on...

The number of elements for any n will be n*(n+1)/2

Note: n is greater than or equal to 0

Write a program that takes input an integer n and returns the required array.

Example Input: 2

Output: 1 1 2

Example Input: 4

Output: 1 1 2 1 2 3 1 2 3 4

For example:

Input	Result
2	1 1 2
4	1 1 2 1 2 3 1 2 3 4

Answer: (penalty regime: 0 %)

```
import java.util.*;
public class Array2
{
    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
        int size = (n*(n+1))/2;
        int[] a = new int[size];
        int c=0;

        for(int i=0;i<n;i++)
        {
            int s=1;
            for(int j=0;j<=i;j++)
            {
                a[c]=s;
                c+=1;
                s+=1;
            }
        }
        for(int i=0;i<size;i++)
        {
            System.out.print(a[i]+" ");
        }

    }
}
```

	Input	Expected	Got	
✔	2	1 1 2	1 1 2	✔

	Input	Expected	Got	
✔	4	1 1 2 1 2 3 1 2 3 4	1 1 2 1 2 3 1 2 3 4	✔

Passed all tests! ✔

Question **3**

Incorrect

Marked out of 25.00

Mr. Jack is now judging arrays! He wants to see if an array can be split. He wants to split it only if the sum of the array on one side is equal to sum of the array on the other side.

For example, in the array: 1 1 1 2 1, it can be split such that 1 1 1 are on one side and 2 1 are on the other side. However the array 2 1 1 2 1 can't be split.

Write a program that accepts input an integer array and prints true if the array can be split, otherwise it prints false.

Note: the array will be of length greater than 1.

Example Input:
5
1 1 1 2 1

Output: true

Example Input:
5
2 1 1 2 1

Output: false

Example Input:
2
10 10

Output: true

Example Input:
6
4 3 1 1 1 4

Output: true

For example:

Input	Result
5 1 1 1 2 1	true
5 2 1 1 2 1	false
2 10 10	true
6 4 3 1 1 1 4	true

Answer: (penalty regime: 0 %)

```
import java.util.*;
public class Array3
{
    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
        int sum = 0;
        int a[] = new int[n];

        for(int i=0;i<n;i++)
        {
            a[i] = sc.nextInt();
            sum += a[i];
        }
        if(sum%2==1)
        {
            System.out.println("false");
        }
```

```
        else
        {
            int maxSum = sum/2;
            int s1=a[0],s2=a[n-1],c=1,d=n-2;

            while(s1<maxSum)
            {

                s1+=a[c];
                c+=1;
            }
            while(s2<maxSum)
            {
                s2+=a[d];
                d-=1;
            }
            if(s1==s2)
            {
                System.out.println("true");
            }
            else
            {
                System.out.println("false");
            }
        }

    }
}
```

	Input	Expected	Got	
✔	5 1 1 1 2 1	true	true	✔
✔	5 2 1 1 2 1	false	false	✔
✔	2 10 10	true	true	✔
✔	6 4 3 1 1 1 4	true	true	✔

Your code failed one or more hidden tests.
Your code must pass all tests to earn any marks. Try again.

Question **4**

Incorrect

Marked out of 25.00

Dr. CooCoo has a new term: reflections.

She says that, in an array, a reflection section are those elements that are:

· Group of continuous elements

· The group appears somewhere in the array in backwards order

You have to find the length of the longest such reflection section.

For example: in the array, 4 5 7 2 1 7 5 4, the longest reflection section is 4 5 7 and it's size is 3.

Write a program that takes input an integer array and prints the length of the largest reflection section.

The Met method has to be inside a Solution class. Please check the code editor for the ideal method definition.

Example Input:
8
4 5 7 2 1 7 5 4

Output: 3

Example Input:
10
10 9 8 7 1 2 7 8 9 10

Output: 4

Example Input:
4
1 2 1 4

Output: 3

Example Input:
9
3 3 3 5 3 3 3 1 1

Output: 7

For example:

Input	Result
8 4 5 7 2 1 7 5 4	3
10 10 9 8 7 1 2 7 8 9 10	4
4 1 2 1 4	3
9 3 3 3 5 3 3 3 1 1	7

Answer: (penalty regime: 0 %)

```
import java.util.*;
public class Array4
{
    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
        int a[]= new int[n];
        for(int i=0;i<n;i++)
        {
            a[i]=sc.nextInt();
        }
        for(int i=n-1;i>=0;i--)
        {
            System.out.print(a[i]+" ");
        }
```

```
    }
}
```

	Input	Expected	Got	
✖	8 4 5 7 2 1 7 5 4	3	4 5 7 1 2 7 5 4	✖
✖	10 10 9 8 7 1 2 7 8 9 10	4	10 9 8 7 2 1 7 8 9 10	✖
✖	4 1 2 1 4	3	4 1 2 1	✖
✖	9 3 3 3 5 3 3 3 1 1	7	1 1 3 3 3 5 3 3 3	✖

Some hidden test cases failed, too.
Your code must pass all tests to earn any marks. Try again.

Show differences