

Sample 1:

I have a number. Should you guess my number, I will say “You guessed my number.” Otherwise, I will say “Lower” when your guess was too high, and I will say “Higher” when your guess was too low.

For example:

<u>my number</u>	<u>your guess</u>	
5	3	Higher
	6	Lower
	4	Higher
	5	You guessed my number.

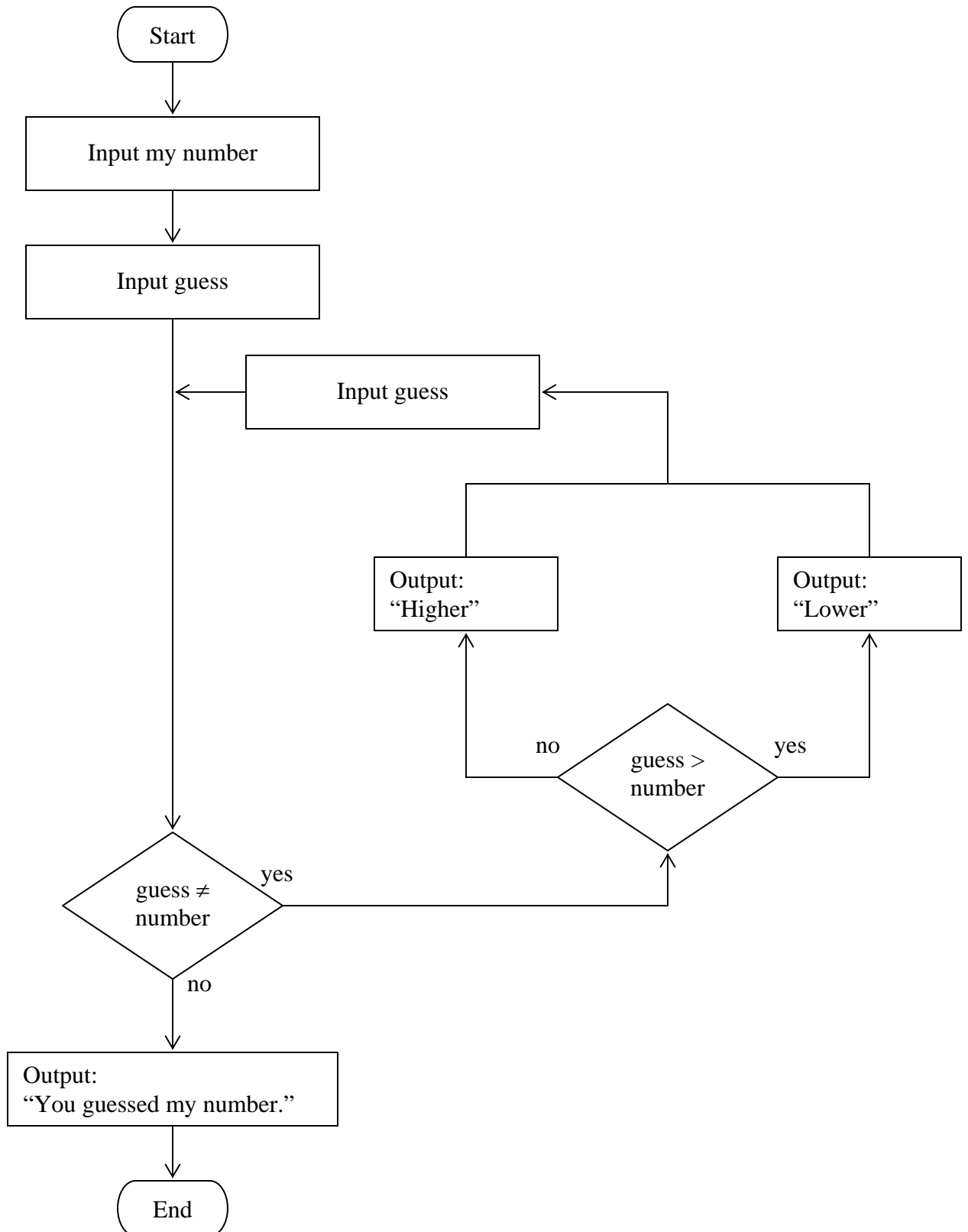
On the following page, there is a flowchart for a program that will take as input the value of my number and the value of your guesses, and that will output my response to each guess.

For Example:

Input: 5	No output.
Input: 3	Output: Higher
Input: 6	Output: Lower
Input: 4	Output: Higher
Input: 5	Output: You guessed my number.

On the page after the flowchart, please convert the program from the flowchart into JAVA.

Flowchart:



Program:

```
public class Sample1
{
    public static void main( String args [] )
    {

    }
}
```

Sample 2:

The triangular numbers are as follows:

1	1
3	1 + 2
6	1 + 2 + 3
10	1 + 2 + 3 + 4
15	1 + 2 + 3 + 4 + 5
etc.	

The series begins with 1 (the first triangular number). To calculate the n^{th} triangular number, n is added to the previous triangular number. For example, the fourth triangular number is calculated by adding 4 to the third triangular number (which is 6), i.e. $10 = (1 + 2 + 3) + 4$.

We need a program that takes a single integer n as input and outputs which triangular number it is or 0 if it is not a triangular number.

For example:

<u>n</u>	<u>output</u>	
15	5	// 15 is the 5 th triangular number
7	0	// 7 is not a triangular number
-1	0	// -1 is not a triangular number

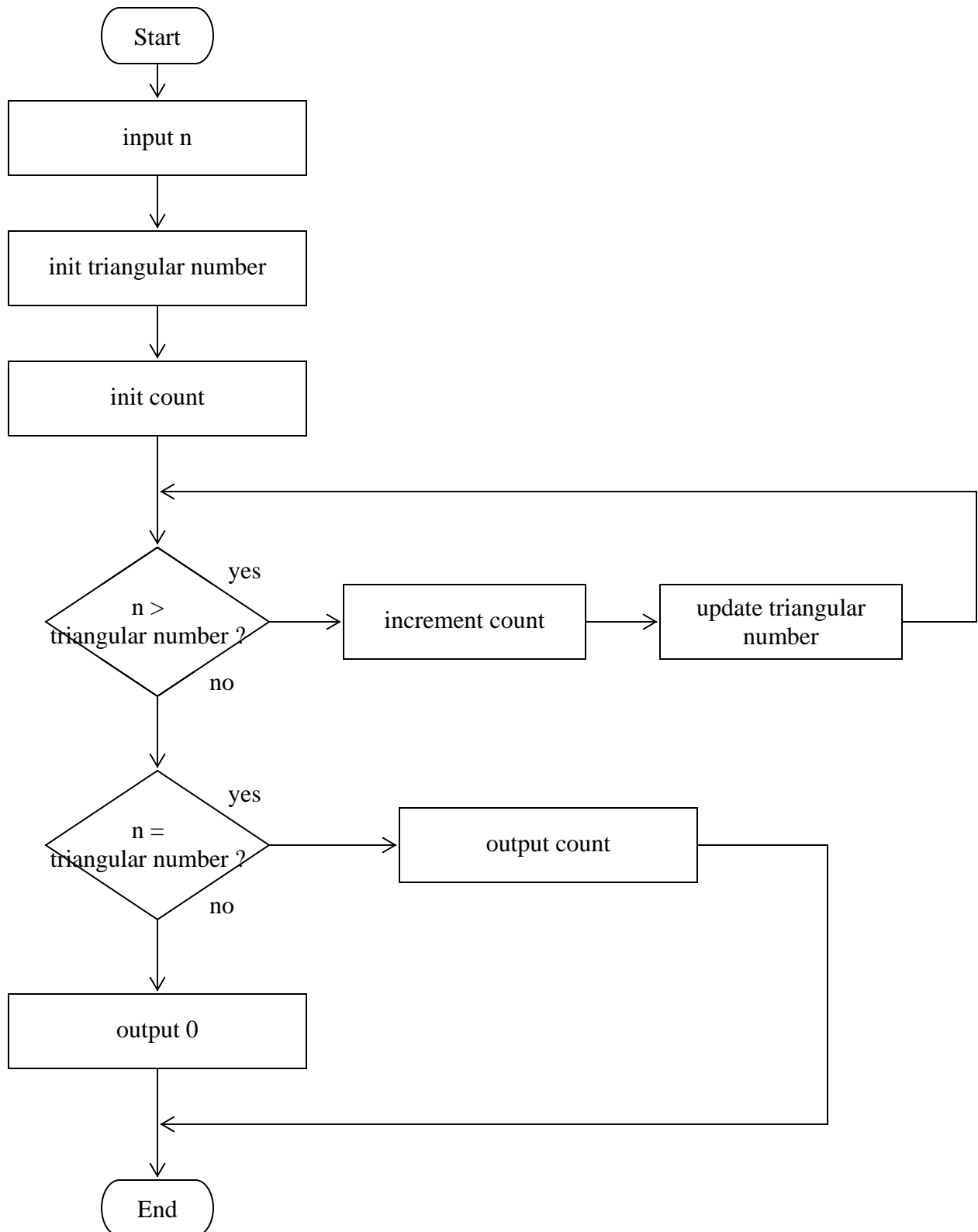
On the following page, there is a flowchart for a program that will take as input a single integer n , and that will output which triangular number it is or 0 if it is not a triangular number.

For Example:

Input: 15	Output: 5
Input: 7	Output: 0
Input: -1	Output: 0

On the page after the flowchart, please convert the program from the flowchart into JAVA.

Flowchart:



Program:

```
public class Sample2
{
    public static void main( String args [] )
    {
```

```
    }
}
```