The Fibonacci Sequence

The Fibonacci sequence appears in nature all around us, in the arrangement of seeds in a sunflower and the spiral of a nautilus for example.

The Fibonacci sequence begins with $fibonacci(0) = 0$ and $fibonacci(1) = 1$ as its first and second terms. After these first two elements, each subsequent element is equal to the sum of the previous two elements.

Programmatically:

- $fibonacci(0) = 0$
- $fibonacci(1) = 1$
- $fibonacci(n) = fibonacci(n-1) + fibonacci(n-2)$

Given n, return the n^{th} number in the sequence.

Example

$n = 5$

The Fibonacci sequence to 6 is $fs = [0, 1, 1, 2, 3, 5, 8]$. With zero-based indexing, $fs[5] = 5$.

Function Description

Complete the recursive function $fibonacci$ in the editor below.

fibonacci has the following parameter(s):

- int n: the index of the sequence to return

Returns

- int: the n^{th} element in the Fibonacci sequence

Input Format

The integer n.

Constraints

- $0 < n \leq 30$

Sample Input

```
STDIN   Function
-----   --------
3       n = 3
```

Sample Output

```
2
```

Explanation

The Fibonacci sequence begins as follows:

$fibonacci(0) = 0$
$fibonacci(1) = 1$
$fibonacci(2) = (0 + 1) = 1$
$fibonacci(3) = (1 + 1) = 2$
$fibonacci(4) = (1 + 2) = 3$
$fibonacci(5) = (2 + 3) = 5$
$fibonacci(6) = (3 + 5) = 8$
...

In the sequence above, $fibonacci(3)$ is 2.

Change Theme Language Java 8

```java
import java.util.*;

public class Solution {

    public static int fibonacci(int n) {

        int answer = 0;

        if (n == 0) {
            answer = 0;
        }

        else if (n == 1 || n == 2) {
            answer = 1;
        }

        else {
            answer = fibonacci(n -1) + fibonacci(n -2);
        }
```

Line: 6 Col: 9

Upload Code as File Run Code Submit Code

Test against custom input

Congratulations!

You have passed the sample test cases. Click the submit button to run your code against all the test cases.

Sample Test case 0
Sample Test case 1
Sample Test case 2

Input (stdin) Download

```
1 5
```

Your Output (stdout)

```
1 5
```

Expected Output Download

```
1 5
```