

Day 9: Recursion 3

Objective

Today, we are learning about an algorithmic concept called *recursion*. Check out the [Tutorial](#) tab for learning materials and an instructional video.

Recursive Method for Calculating Factorial

$$factorial(N) = \begin{cases} 1 & N \leq 1 \\ N \times factorial(N - 1) & otherwise \end{cases}$$

Function Description

Complete the *factorial* function in the editor below. Be sure to use recursion.

factorial has the following paramter:

- *int n*: an integer

Returns

- *int*: the factorial of *n*

Note: If you fail to use recursion or fail to name your recursive function *factorial* or *Factorial*, you will get a score of 0.

Input Format

A single integer, *n* (the argument to pass to *factorial*).

Constraints

- $2 \leq n \leq 12$
- Your submission must contain a recursive function named *factorial*.

Sample Input

3

Sample Output

6

Explanation

Consider the following steps. After the recursive calls from step 1 to 3, results are accumulated from step 3 to 1.

1. $factorial(3) = 3 \times factorial(2) = 3 \times 2 = 6$
2. $factorial(2) = 2 \times factorial(1) = 2 \times 1 = 2$

3. *factorial*(1) = 1