

Recursion Review

Recursion

Recursion is when a method calls itself.

```
public static void neverEnd()  
{  
    System.out.println("This is the method that never ends!");  
    neverEnd();  
}
```

This is an infinite recursion.

Recursion reduces a problem to smaller, similar, problems that can be solved more easily than the larger problem.

Google and recursion: Go to google and search recursion. What do you notice?



Recursion cases

A recursive function has a conditional for dealing with two different cases:

- **The recursive case:** The function calls itself to solve a smaller problem then use that solution to solve the complete problem.
- **The base case:** The unction recognizes the simplest situations and completes the computation without calling itself.

Both cases are important. Since working out the recursive case usually takes a lot of thinking, programmers often forget about the simple base case. When that happens, a recursive function will call itself repeatedly in an infinite recursion and the function will never complete.



The Call Stack

Recursive functions use something called “the call stack.”

When a program calls a function, that function goes on top of the call stack (like stack of books)

Things are added one at the time.

To take something off, the top item is removed first.



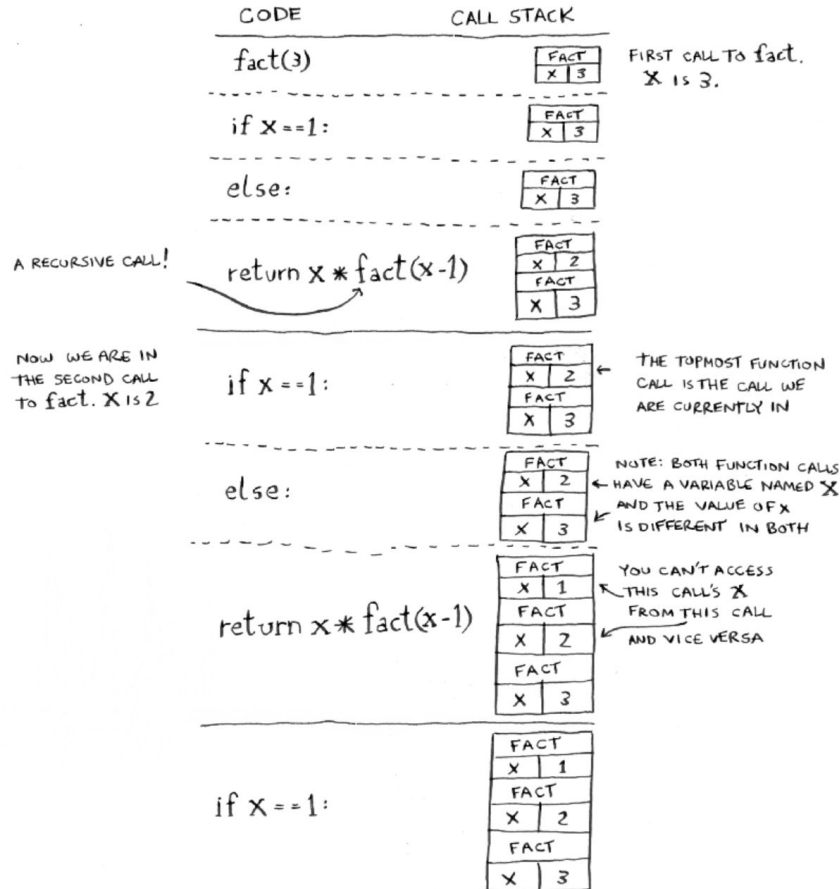
Factorial Function Example

`factorial(3)` is written as $3!$ and it is defined like this: $3! = 3 * 2 * 1$. Here is a recursive function to calculate the factorial of a number:

```
function fact(x) {  
  if (x == 1) {  
    return 1;  
  } else {  
    return x * fact(x-1);  
  }  
}
```

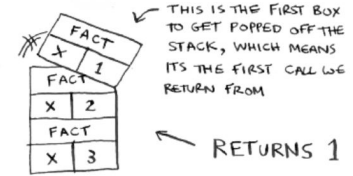


Call Stack - fact(3)



WOW, WE MADE THREE CALLS TO fact, BUT WE HAD NOT FINISHED A SINGLE CALL UNTIL NOW!

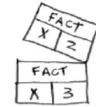
return 1



THIS IS THE FUNCTION CALL WE JUST RETURNED FROM

return x * fact(x-1)

X IS 2



← RETURNS 2

return x * fact(x-1)

X IS 3

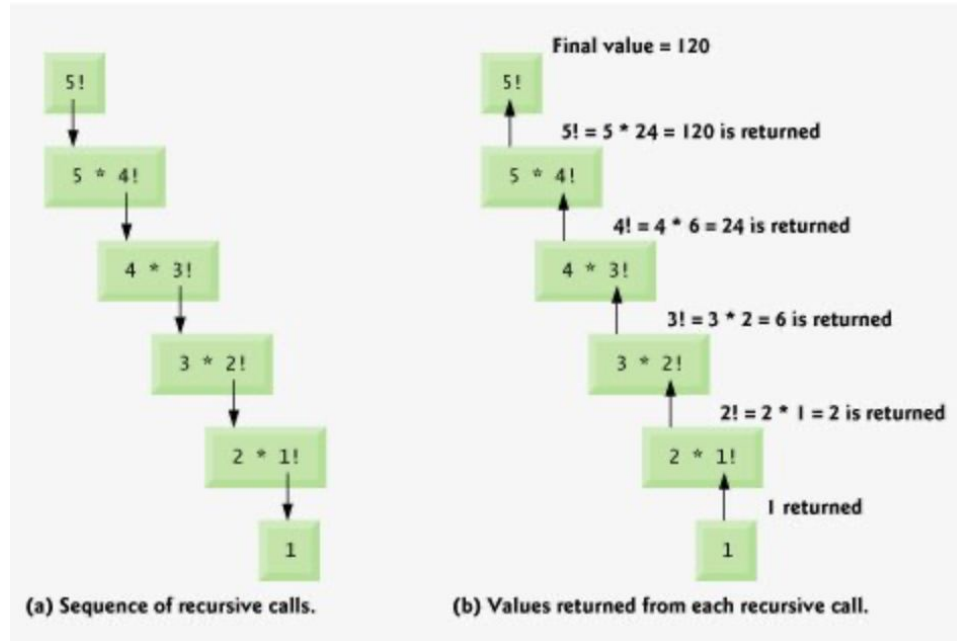
THIS CALL RETURNED 2



← RETURNS 6

Image credit: Adit Bhargava

Recursion - Factorial



Practice

https://github.com/novillo-cs/apcsa_material/blob/main/lessons/recursion/practice.md

