

1. Suppose I run the following method: `public static int b(int n) { return b(n); }`. What will happen?

It would Call the method B(n) infinitely.

2. Write a recursive method that obtains the maximum element of an array of integers, starting from the tail. Give the stack trace for your own example array of at least size 4.

Example Array: 4,3,16,10

`maxInt(arr, 0, 3)`

Calls `maxInt(arr, 10,2)` because 10 is more than 0, count is reduced by 1

Calls `maxInt(arr, 16,1)` because 16 is more than 10, count is reduced by 1

Calls `maxInt(arr, 16, 0)` because 16 is more than 3, count is reduced by 1

Returns 16, because 16 is more than 4

3. Write a recursive method that sums all even or negative numbers in an array of integers. Give the stack trace for your own example array of at least size 4

Example Array: 4,-3,16,9

`evenNegs(arr, 3)`

Calls `evenNegsSum(arr,2)` without adding anything since 9 is neither negative nor even

16 + `evenNegsSum(arr,1)` since 16 is even

-3 + `evenNegsSum(arr,0)` since -3 is negative

+ 4 since 4 is even

4 + (-3) + 16 = 17, so the ultimate return is 17

4. Write a recursive method that removes all consecutively occurring letters from a string of fixed size. E.g. "AAAbbCCCC" becomes "AbC"

5. Bonus: Question C-5.16 from the book. (Chapter 5) This is the classic Towers of Hanoi problem.

4. Written Pseudocode is fine for this question

The Psuedocode

FUNCTION `hanoi(disks, source, dest, space)`:

IF (`disks == 1`) THEN:

Move the top disk from source to dest

ELSE:

`hanoi(disks - 1, source, space, dest)` -> send disk 1 to the middle space

Move top disk from source to dest -> send disk 2 to the destination

`hanoi(disks - 1, space, dest, source)` -> send disk 1 from middle space to destination