Algorithmic Complexity

Stairmaster

Intermediate - java

Write a function, `stairmaster(n)`, that will compute the number of ways to climb a flight of `n` steps, taking `1`, `2`, or `3` steps at a time.

Take the example of climbing `n = 4` steps. There are seven different ways one can climb four stairs using 1, 2, or 3 steps at a time: [1,1,1,1] [2,1,1] [1,2,1] [1,1,2] [2,2] [1,3] [3,1].

Make sure to find all permutations, not combinations, as the order matters. Climbing one step then two steps is different from climbing two steps then one step.

This challenge was reported to have been asked at interviews with Facebook, as well as right here at Codecademy! If you’ve covered the material in Pass the Technical Interview with Java or an equivalent, you should be able to solve this challenge. If you have trouble, try refreshing your knowledge there first.