Staircase

May 18, 2020

0.0.1 Problem Statement

Suppose there is a staircase that you can climb in either 1 step, 2 steps, or 3 steps. In how many possible ways can you climb the staircase if the staircase has `n` steps? Write a recursive function to solve the problem.

Example:

- `n == 1` then `answer = 1`
- `n == 3` then `answer = 4` The output is `4` because there are four ways we can climb the staircase:
  - 1 step + 1 step + 1 step
  - 1 step + 2 steps
  - 2 steps + 1 step
  - 3 steps
- `n == 5` then `answer = 13`

0.0.2 Exercise - Write a recursive function to solve this problem

```
In [4]: """
        param: n - number of steps in the staircase
        Return number of possible ways in which you can climb the staircase
        """
        def staircase(n):
            '''Hint'''
            # Base Case - What holds true for minimum steps possible i.e., n == 0, 1, 2 or 3? Re

            # Recursive Step - Split the solution into base case if n > 3.

            pass
```

Show Solution

```
In [ ]: def test_function(test_case):
            n = test_case[0]
            solution = test_case[1]
            output = staircase(n)
```

```
            if output == solution:
                print("Pass")
            else:
                print("Fail")
```

In [ ]:
```
n = 3
solution = 4
test_case = [n, solution]
test_function(test_case)
```

In [ ]:
```
n = 4
solution = 7
test_case = [n, solution]
test_function(test_case)
```

In [ ]:
```
n = 7
solution = 44
test_case = [n, solution]
test_function(test_case)
```