**CTA 5: Recursive**

Jordon Paynter

Department of Computer Science, Colorado State University Global

CSC 400: Data Structures and Algorithms

Professor Santosh Kumar Gottipamula

October 15, 2023

Pseudocode

class RecursiveFactorials:

method factorialRecursive(n):

if n == 0:

return 1

else:

return n \* factorialRecursive(n - 1)

End method

method factorialAlternative(n):

if n <= 1:

return 1

else:

return n \* factorialAlternative(n - 1)

End method

End Class

class ReversingArray:

method reverseArrayPortion(arr, startIndex, endIndex):

if startIndex < endIndex:

swap arr[startIndex] with arr[endIndex]

recursively call reverseArrayPortion(arr, startIndex + 1, endIndex - 1)

End method

End Class

class JunitTestingOfCTA5:

test testFactorialRecursive():

assert factorialRecursive(0) == 1

assert factorialRecursive(1) == 1

assert factorialRecursive(4) == 24

assert factorialRecursive(5) == 120

End test

test testFactorialAlternative():

assert factorialAlternative(0) == 1

assert factorialAlternative(1) == 1

assert factorialAlternative(4) == 24

assert factorialAlternative(5) == 120

End test

test testReverseArrayPortion():

arr = [1, 2, 3, 4, 5]

reverseArrayPortion(arr, 1, 3)

assert arr == [1, 4, 3, 2, 5]

arr2 = [9, 7, 5, 3, 1]

reverseArrayPortion(arr2, 0, 4)

assert arr2 == [1, 3, 5, 7, 9]

arr3 = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21]

reverseArrayPortion(arr3, 5, 15)

assert arr3 == [1, 2, 3, 4, 5, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 17, 18, 19, 20, 21]

End test

End Class

For the Big-O analysis:

The factorial methods have an O (n) time complexity because they perform n multiplications in the recursive calls.

The reversing of the array method has a time complexity of O (N/2). N is the number of elements in the array portion you want to reverse. It is divided by two because the function only needs to go through half the amount of N to complete the task. As far as reversing the whole.

**Figure 1.**

Display test receiving Green Check Marks by line numbers!

A screenshot of a computer program

Description automatically generated

**Figure 2.**

Displaying the Test being ran and each test that was ran with passing checks!A screenshot of a computer

Description automatically generated