COSC1437 Programming Lab & Assignment (Fall 2016)

Iteration vs. Recursion

Topics

- In this lab, we practice implementing several well-known functions in recursive or/and non-recursive (i.e., iterative) version(s).

Activities

- You are given a draft of the java code, “Recursion.java”, for this assignment. This given code is not complete and thus if you are trying to compile it as it is, you will get ~100 compilation errors.
- Your mission is to implement all the required method in a java code, called “RecursionAnswer.java”. First, you rename the given java code (“Recursion.java”), fix the errors and complete main() method to get the exactly the same running example shown in the next page.

  (1) Add “public” access modifier to every method.

  (2) Implement every method to be a class method.

  (3) Correct all the errors (Hint: change method’s signature).

  (4) Implement main() to get the running example.

- Refer the following UML diagram for the details of required program components (including the class name, data fields, and methods) for this lab.

RecursionAnswer
+fact1 (n: int): int **+fact2 (n:int): int** **+writeBackward1(str: String, size: int): void** **+writeBackward2(str: String, size: int): void** **+writeBackward3(str: String, size: int): void** **+writeBackward4(str: String, size: int): void** **+sum1(int n): int** **+sum2(int n): int** **+rangeSum1(arra)y: int[], start: int, end: int): int** **+rangeSum2(array: int[], start: int, end: int): int** **+fib1(n:int): int** **+fib2(n:int): int** **+fib3(n:int): int** **+acker(m: int, n: int): int** **+gcd(a: int, b: int): int** **+c(n: int, k:int): int** **+main(args: String[]): void**

Running Example *(identical to the actual results you will get)*

```
Test fact(int n):
    fact1(10) = 3628800
    fact2(10) = 3628800
Test WriteBackward(String str, int size):
    writeBackword1("Your name", 9) = eman ruoY
    writeBackword2("Your name", 9) = eman ruoY
    writeBackword3("Your name", 9) = eman ruoY
    writeBackword4("Your name", 9) = eman ruoY
Test sum(int n):
    sum1(10) = 55
    sum2(10) = 55
Test rangeSum(int array, int m, int e)
    rangeSum1(array, m, e) = 49
    rangeSum2(array, m, e) = 49
Test fib(int n):
    fib1(1) = 1
    fib2(1) = 1
    fib3(1) = 1
    fib1(2) = 1
    fib2(2) = 1
    fib3(2) = 1
    fib1(10) = 55
    fib2(10) = 55
    fib3(10) = 55
Test acker(int m, int n):
    acker(0, 10) = 11
    acker(3, 0) = 5
    acker(3, 10) = 8189
Test gcd(m, n):
    gcd(3, 10) = 1
Test c(n, m):
     c(10, 3) = 120
Press any key to continue…
```

BONUS (Optional)

- N/A

What to Hand in

- Turn your program (i.e., “RecursionAnswer.java”) in via Blackboard.