CSI 31 LECTURE NOTES (Ojakian)

Topic 17: Practicing Recursion

OUTLINE
(References: 11.3, 11.4)

---

1. Evaluating Recursion Functions

   (a) What do the functions defined in the accompanying file output?
   "Expand" to check that works.

   (b) i. Illustrate recursive function with NO return: Just forward pass.
   ii. Illustrate recursive function WITH return: Forward and backwards pass.

2. Writing Recursion Functions

   PROBLEM 1. *Calculate the number of X's in a string using recursion (without using the* `count` *function).*

   PROBLEM 2. *Write a recursive function that takes a string as input and returns the number of vowels. There is a good way and a bad way! (like with Binary Search)*

   PROBLEM 3. *Write a recursive function that takes a positive integer n as input. The function outputs the sum of the cubes of 1, 2, ..., to n. For example, on input 3, it should output 32. The function may not use: Loops, lists, sets, or dictionaries. Write it so that it does some error checking: While it can take any number as input, for any number besides a positive integer, it returns 0.*

   PROBLEM 4. *Do the last program again, but now take two inputs a and b, returning the sum of the cubes inbetween a and b (including a and b). For example, on inputs -2 and 3, return 27. Consider various ways to do this ...*

   PROBLEM 5. *Write a recursive function that takes a string as input and returns True if the string alternates '0' then '1', and False otherwise. For example F('0101010') return True, while F('010110') returns False. No loops allowed.*