CptS 121 - Program Design and Development

Lab 12: More with Recursion

Assigned: Week of April 14, 2014
Due: At the end of the lab session

I. Learner Objectives:

At the conclusion of this programming assignment, participants should be able to:

- Utilize recursion to solve problems

II. Prerequisites:

Before starting this programming assignment, participants should be able to:

- Utilize output parameters and pointers in a C program
- Apply the dereference or indirection C operator
- Declare strings in C
- Apply library functions found in <string.h>
- Implement array notation or pointer arithmetic to manipulate strings
- Distinguish between character arrays and strings in C
- Pass arrays into functions
- Initialize arrays using an initializer list
- Construct loops to traverse through arrays

III. Overview & Requirements:

This lab, along with your TA, will help you navigate through applying recursion to solving problems.

Labs are held in a "closed" environment such that you may ask your TA questions. Please use your TAs knowledge to your advantage. You are required to move at the pace set forth by your TA. Please help other students in need when you are finished with a task. You may work in pairs if you wish. However, I encourage you to compose your own solution to each problem. Have a great time! Labs are a vital part to your education in CptS 121 so work diligently.

Tasks:

1. **Skip this problem if you solved it for lab 11.** Complete Project 5 in Chapter 10 on page 563. Write a function that accepts an 8 by 8 array of characters that represents a maze. Each position can contain either an X or a blank. Starting at position (0, 1), list any path through the maze to get to location (7, 7). Only horizontal and vertical moves are allowed. If no path exists, write a message indicating there is no path.

   Moves can be made only to locations that contain a blank. If an X is encountered, that path is blocked and another must be chosen. Use recursion.

2. Write a function which returns the sum of the digits of an integer value. You must use recursion. Example sum (765) = 18, sum (16978) = 31, etc.

3. Write a function which recursively reverses a string.

4. Tower of Hanoi: A very popular mathematical game or puzzle is referred to as the Tower of Hanoi. The idea behind the game is to find an efficient method for moving disks between *three* posts. Each disk has a different diameter, but all of them can be placed on the available posts. The goal of the game is to move all of the disks from one post to another according the following rules:

    1. Only one disk may be transferred at a time
    2. Only the top disk on any post may be accessed at a given time
    3. No disk may be placed on top of a smaller disk at any point

    At the start of the game, all of the disks must originally be placed such that the largest disk is on the bottom of the stack of one post, and the smallest is on the top of the stack on the same post. The disks should form a cone shape. Write a program to simulate the Tower of Hanoi game. For each move print the post number (1 – 3) from which the disk is taken, the diameter of the disk, and the resulting post on which the disk is placed. Also, show the current diameter of the disks on each post. You must use recursion to solve this problem! Initially start with three disks in your game. Note: if you visit http://www.mazeworks.com/hanoi/, you will find an animation of how Tower of Hanoi should run.

IV. Submitting Labs:

- You are not required to submit your lab solutions. However, you should keep them in a folder that you may continue to access throughout the semester. You should not store your solutions to the local C: drive on the Sloan 353 machines. These files are erased on a daily basis.

V. Grading Guidelines:

- This lab is worth 10 points. Your lab grade is assigned based on completeness and effort. To receive full credit for the lab you must show up on time and continue to work on the problems until the TA has dismissed you.