

## Lab3 - CIT315

### Goals

The primary goal of this project is to write, test and complete a *c* program which uses *functions* and *recursion*. It is critical that you know how to program and use recursion. While you are given the code for some of these functions, make sure you know how the functions execute! There will be some functions that you must write from scratch. All functions will be called by a numbered user menu, prompting the user for the correct inputs for each one. Then, you will print the output in a neatly formatted and understandable manner. The menu must ask the user if they want to run more functions and also ask them if they want to quit the program, and act accordingly.

### Specifications

For each of these, write a simple function. In some cases the code is given to you. Just place it in context of a program. **You will have at minimum 6 functions, in your program.** The main function and the 5 functions you will write. ***You must use function prototypes!*** Then, **create a basic menu system that allows a user to select which one of the functions they want to execute.** Once the user makes their choice, **the program must ask them for the correct parameters.** After the parameters are entered, the program will execute and print the results to to command line. Each function must be properly formatted, with a prototype.

When the program is finished, return the number of functions that were executed in your program to the operating system. For example, if you executed all 5 functions, the return would be 5. If you executed all of them two times, the return value would be 10.

### Function 1 - Factorials

**Task:** Write a function in your *C* program based on the code given below.

The factorial function is an example of direct recursion and is defined by:

$$n! = \begin{cases} 1 & \text{if } n = 0 \\ n(n-1)! & \text{if } n > 0 \end{cases}$$

The code to use is:

```
int factorial(int n)
{
    if (n==0)
        return 1;
    else
        return n * factorial(n - 1);
}
```

Output: Print out a message and the input and output numbers.

## Function 2 - Fibonacci numbers

**Task:** Use the following definition to create a *C* function which takes a *int* as a parameter and gives the Fibonacci sequence for that number. (Hint: the function header will be: *int fibonacci(int n)* )

The Fibonacci function is defined by:

$$F_n = \begin{cases} 0 & \text{if } n = 0 \\ 1 & \text{if } n = 1 \\ F_{n-1} + F_{n-2} & \text{if } n > 1 \end{cases}$$

output: Print out the list of Fibonacci numbers.

## Function 3 - Towers of Hanoi

**Task:** Copy the following function into your program and complete the code with a *Printf* statement that prints all moves.

```
void runHanoi(int n, char x, char y, char z)
{
    if (n == 1)
    {
        //printf...
    }
    else
    {
        runHanoi(n-1,x,z,y);
        runHanoi(1,x,y,z);
        runHanoi(n-1,y,x,z);
    }
}
```

Output: Print all of the moves made to the discs on the pegs.

## Function 4 - Reversing a string

**Task:** Use recursion to reverse a string, by changing the order of its characters. You must use direct recursion for this and not any canned functions!

Output: Print the input and output strings.

## Function 5 - Conversion to ASCII

**Task:** This problem has two parts. First, take an input string of ASCII characters and convert it to a binary string of 1's and 0's, representing the textual string. For example, if the user input the

string "ac" the 'a' ASCII character has binary equivalent of "01101001" and the 'c' character has a binary "01100011".

## Rubric

The scoring for this assignment is as follows, with 25 points maximum:

- Overall program structure, such as looping and menu - 5 points
- Functions 1 to 2 - 2 point each
- Functions 3 to 4 - 3 points each
- Function 5 - 5 points
- returning the correct value to the operating system - 2 points
- Name, class number and brief description at the top in comments - 3 points

## Submission

Complete the code and submit the project file(s) by the required date and time to Brightspace.