Nick Tansino

Prof. Hervé

CSC 412 - 0001

28 September 2020

Prog02 Report:

Prog02:

For this assignment, I decided to complete these tasks by breaking each one down into its own function. That being said, there were three total functions that I created throughout this assignment (main(), fibonacci(), and validate()). The validate function takes ‘argc’, ‘F1’, ‘F2’, ‘n’ (all as integers), and the name of the executable file (execName) as arguments. Its main task is to verify that the user’s data is valid before allowing the fibonacci function to work with the input. Given that the data is valid, the fibonacci function recursively prints the correct answer to the user, otherwise the user will receive one of two error messages. The last two arguments in the fibonacci function are used to assure formatting is correct throughout the printing process. The main difference between Prog02Part1.c and Prog02Part2.c is the implementation of the main function. In Part 1, the main function converts ‘F1’, ‘F2’, and ‘n’ from ASCII to integer values, and then validates said integers. If the input is valid, the main function then executes the fibonacci function. In Part 2, it does the exact same thing, but also checks that the list size is correct (divisible by 3), and then executes the validate and fibonacci functions as previously described.

Script02:

For the bash script, I handled mainly everything I could with conditional statements. For instance, the first line of code is an ‘if’ statement that checks for the correct number of arguments. If the script is given too many arguments, the user is prompted with an error sooner, rather than if error handling was done near the end. Given that the input is correct, the rest of the script travels to the given directory, searches every file in the directory, and adds files with the correct extension to a previously created array. With every file that is added to the array, a count variable is incremented, which is later used to determine how many files will be printed.

Limitations:

While I have not come across any limitations that affect the functionality of my program(s), the only part of the assignment that I wasn’t able to implement was the array at the end of Prog02Part2.c, mainly due to the fact that I was having trouble keeping track of array allocation and recursion all at once.