**Programming Assignment #1**

**CSCI 4534**

**Date Assigned**: January 31, 2018

**Date Due**: Tuesday, February 13th, 11:59pm to Blackboard

This assignment is meant to:

1. Introduce students to writing programs in C using the UNIX platform
2. Provide an opportunity to write code similar to that of an operating system’s memory manager

This assignment requires the analysis, implementation, testing and documentation of a C program that uses C on the UNIX server sapphire.

**Problem:**

Implement a program in which the user enters a sequence of memory requests. The program will determine the starting address, the final address, and the amount of internal fragmentation for a memory with fixed-size partitions. Store the information in a struct that contains 2 integers and an array. The first integer designates the sizes of all fixed –size memory partitions, the second integer designates the number of partitions, and the array holds the sequence of memory requests as though coming from several users. Your program will assign the memory requests to the partitions in order using **first fit**, fitting as many requests as possible. After assignments are made, your program will report the first address assigned and the last address assigned inside each partition. For each partition, report internal fragmentation. Report the number of unassigned partitions and the beginning address of each.

**Submit your work to Blackboard as follows:**

* Save your C program (possibly using Notepad++), and name it with the .c extension.
* Create a folder named lastNameFirstInitial\_CSCI4354\_Program# e.g. HallS\_CSCI4354\_Program1
* Place the following into the folder

1. Your **.c** file which is **documented** following the Programming Guidelines. Clearly indicate the name of your program in the asterisks along the top of your preface.
2. In your documentation, add an **estimated** and **actual** time spent for each of the following: design, implementation, and testing.
3. A separate **ReadMe** file that explains how to execute your program.
4. **Input values** and **Output values** with the screenshots for 2 executions of your program.

* Zip the folder: Right click on the folder and send to compressed file

For video instructions: <http://youtu.be/exRJS94Fnfc>

**Late Programs**: Programs are due to Blackboard at 11:59 pm on the due date. Programs submitted late will have 20 points deducted per day for a maximum of two days.

**Back up your files often!**

Homework for Thursday, September 21st

Turn in a **preface** for your program that contains all parts designated by the Programming Guidelines, including the pseudocode/algorithm for solving this problem.