

Erik Larsen

Southern New Hampshire University

Algorithm Narrative

This algorithm update is an update of the final project for my first class. It is a program that uses a credential file that stores user information and compares the input data to the file. After comparing, the user is either allowed to log in and returned file information from the data assigned to their role or asked to try again up to 3 tries before they are kicked.

I used this program to show my ability to create algorithms to speed up the process of searching through user credentials. This was done in the form of a quicksort algorithm which would alphabetize the credentials by username and keep all associated information in the same row. Then a binary search algorithm was used to find user information more quickly by searching through the sorted file.

I did meet my main goal of showing my ability to utilize algorithms to speed up processing. Quicksort and binary sort are two algorithms that are known to be able to exponentially reduce the amount of comparisons that the program must compute. Using this instead of a standard linear sort gains more value the longer the list of users becomes.

Enhancing this artifact, I remembered how confusing algorithms can get when trying to work through nested loops and ifs etc. It can be quite a struggle to keep the whole process in track in my head, so I like to use visual aids to understand decisions flow through the process. The only real issue I had come with some bug catching in the process. Fortunately, after a few look throughs I realized it was a simple problem of swapping less than and equal to signs. Even though it is a quick fix, it goes to show coding is extremely specific and you always must keep your eye on the details. When incorporating feedback, I updated file paths of the credentials and

user files to be addressed with relative paths vs absolute paths. This allows for other users to run the program or for me to use a program on a different machine.