Lab Report (Basic Data Structures and Algorithms)

I/we the undersigned, promise that the submitted lab report is/are my/our own work. While I/we was/were free to discuss ideas with others, the work contained is my/our own. I/we recognize that should this not be the case; I/we will be subject to penalties as outlined in the course syllabus.

(By typing in your name below, you agree to Academic Integrity and honesty)

Name: **Nero Hamidi**

Red-Id: **827723033**

Reflection:

In a short paragraph explain your learnings and understanding on basic data structures and algorithms

Explain

- Hashing
- Divide and conquer
- Big OH (O) notation
- Logarithmic time complexity
- Stack

Data Structures and Algorithms both help programmers complete their code more efficiently. Data structures are how computers store data, different data structures provide different functionality, which makes it important to select the right data structure for whatever task you wish to complete. Algorithms are programs that you provide to the computer which has a set of steps for it to complete. Algorithms take up different amounts of time and space within the computer depending on the way they are written and the task at hand; it's important to understand the different algorithms to select the right one for whatever task you are looking to complete.

Hashing is a system where data is mapped onto a specific value so it can be accessed later; this is helpful in terms of easily storing and retrieving data. Divide and Conquer is a problem-solving strategy where the goal is to break up a larger collection of data into smaller chunks to make searching and sorting easier. This helps algorithms in becoming more efficient and can be seen in algorithms such as merge sort. Big O (O) notation is a way to denote the run time complexity of a program. The notation reflects the worst-case scenario for how long it would take to complete a particular program. For example, when searching for a value in an array length N, the worst case is that it searches every value in the array; therefore the Big O (O) notation is O(N). Logarithmic time complexity can be seen in such algorithms as Binary sort. It is when the running time of the program increases logarithmically with an increase in the input; the notation would be O(log(n)). A stack is a data structure that follows the last in first

out rule. This means that the last thing you enter into a stack will be the first thing you retrieve when calling upon it.