Zheen H. Suseyi

CS 300

SNHU

**Evaluation**

**Vector:** Vectors increase in memory usage the more elements the vector contains so for bigger projects vectors may not be ideal. The advantages of a vector include being very easy to implement into our program and the ability to sort our vector. The downsides of vectors is that memory usage may become a huge problem and inserting and deleting elements can have a worst possible runtime of O(n).

**Assuming n courses:**

Reading courseInformation file: O(n)

Reading each line: O(n)

Creating course objects out of the lines: O(n)

Worst Case: O(n)

**Hash Table:** Hash Tables are similar to vectors in which memory usage increases due to the number of elements. The advantages of a Hash Table are extremely fast insertion, deletion, and search times for elements contained inside the Hash Table. The downsides of Hash Tables is the inability to sort a Hash Table along with very poor runtimes if the table gets too large. Hash Tables also may result in collisions if multiple elements are assigned to the same key.

**Assuming n courses:**

Reading courseInformation file: O(n)

Reading each line: O(n)

Creating course objects out of the lines: O(1)

Worst Case: O(n)

**Binary Search Tree:** Binary Search Trees require the most memory out of the three data structures but comes with many upsides. The advantages of Binary Search Trees is that it is the most effective for insertion, deletion, searching, and sorting. The downsides of Binary Search Trees include being slow when the tree gets too large, poor memory usage, and may be difficult to implement into a program. If the tree is unbalanced, there can be worst case runtimes of O(n) for all operations which is a disaster.

**Assuming n courses:**

Reading courseInformation file: O(n)

Reading each line: O(logn)

Creating course objects out of the lines: O(logn)

Worst Case: O(n)