**Evaluation.**

Program upon evaluation will need to open the program and load data from a CSV file, the program will need to store the data attributes. The file will then need to check for the course’s three main attributes such as course ID, the name of the course, and if it’s a prerequisite.

Then the course data will be stored in objects via three possible structures, which are either a linked list binary tree or a hash table. Then displays a message to the user to print a course list, print the course and sort the course list.

Each line that is executed can have different runtimes for the linked list it will take the longest time to run each line, while the hash table and Binary Tree structures have keys and they all can be used to directly access a specific section of the data which lowers the run time.

This applies to function and nested loop calls which can add more time to the program.

As stated earlier the advantages and disadvantages of vectors, hash tables, and tree structures can affect how a program runs and is understood.

|  |  |  |
| --- | --- | --- |
| Structure | Disadvantages | Advantages |
| Vectors | Long Run time O(g(n)) as worst case | Simpler to understand. |
| Hash tables | Can have a collision.  Adding to run time O(n) | Quicker Run times O(1) best case |
| Binary trees. | More complex to implement sorting functions | The best case is O(1) and the worst case is better than Hash table O(log n) |

Based on the analyzed data I will try using BTS as their Big O analysis will have the best run time complexity, If the complexity is too difficult I will attempt Hash tables as a secondary structure and the last structure that can be implemented with the worst runtime is the vectors structure list.