**Evaluation**

|  |  |  |  |
| --- | --- | --- | --- |
| Vector Code | Line Cost | Times Executed | Total Cost |
| Open file | 1 | 1 | 1 |
| Read data | 1 | 1 | 1 |
| Check course title | 1 | n | n |
| Check course number | 1 | n | n |
| CreateCourseObj | 1 | 1 | 1 |
| Create variables | 1 | 1 | 1 |
| Search for Course | 1 | n | n |
| Print course info | 1 | n | n |
| Store data in structure | 1 | 1 | 1 |
|  |  | Total Cost | 4n + 5 |
|  |  | Runtime | O(n) |

|  |  |  |  |
| --- | --- | --- | --- |
| Hash Table Code | Line Cost | Times Executed | Total Cost |
| Open file | 1 | 1 | 1 |
| Read data | 1 | 1 | 1 |
| Check course title | 1 | n | n |
| Check course number | 1 | n | n |
| CreateCourseObj | 1 | 1 | 1 |
| Create variables | 1 | 1 | 1 |
| Search for Course | 1 | n | n |
| Print course info | 1 | n | n |
| Store data in structure | 1 | 1 | 1 |
|  |  | Total Cost | 4n + 5 |
|  |  | Runtime | O(n) |

|  |  |  |  |
| --- | --- | --- | --- |
| Tree Structure Code | Line Cost | Times Executed | Total Cost |
| Create Variables for left & right nodes | 1 | 1 | 1 |
| Create Root = null | 1 | 1 | 1 |
| Create Course Variables | 1 | 1 | 1 |
| Open File | 1 | 1 | 1 |
| Check for course title | 1 | n | n |
| Check for course number | 1 | n | n |
| IF root != null | 1 | n | n |
| Else add course info to right node | 1 | n | n |
| Print results | 1 | 1 | 1 |
|  |  | Total Cost | 4n + 5 |
|  |  | Runtime | O(n) |

After working with these 3 data types each has its advantages and disadvantages. Vectors are not a fixed structure and can be expanded upon if needed. However, if the data is too large traversing gives a constant time of O(n). Hash Tables work well with large sets of data. The downside of hash tables is that they are hard to implement. Tree structures are easy to implement and do not require much space as they are simply nodes with roots with a left and right connection. Tree structures can have long traversal due to the height of the trees, thus O(h).

From these 3 data structures the one I would recommend using is a hash table. While a hash table may run slower, I find it is more easily organized and can handle larger amounts of data. Hash tables can also be updated later to add any additional information required of the program.