Runtime Analysis

The cost per line of code and the number of times a line will execute must be considered in order to determine the pseudocode's worst-case running time for reading the file and creating course objects. Assume the data structure has n courses stored in it.

|  |  |  |  |
| --- | --- | --- | --- |
| Code | Line Cost | # Times Executed | Total Cost |
| For all courses | 3 | 1 | 4 |
| If the course is the same as courseNumber | 2 | 8 | 10 |
| Print out the courses information | 1 | 1 | 2 |
| For each prerequisite of the course | 4 | 8 | 12 |
| Print the prerequisite course information | 2 | 6 | 8 |

|  |  |
| --- | --- |
| Total Cost | 145 |
| Runtime | 0 |

1.Vector

The worst-case running time for adding an element to the end of a vector is O(1), but if the vector needs to resize itself to make room for the new element, the worst-case running time can be O(n). The worst-case running time for adding n elements to a vector is thus O(n^2).

Advantages:

Accessing elements is quick because they are kept together in memory.

It's effective to add elements to the end of a vector.

Disadvantages:

Inserting or deleting elements in the middle of a vector can be slow.

It may be costly to resize a vector.

2. Hash table:

When the keys are distributed evenly by the hash function, inserting an element into a hash table runs in worst-case O(1) time. The worst-case running time, on the other hand, would be O(n) if all the keys had the same hash value.

Advantages:

On average, retrieving and inserting elements is quick.

A dynamic size adjustment is possible for the hash table.

Disadvantages:

Operations may be slowed down by collisions in hashing.

Elements are not kept in the correct order.

3. Tree:

The worst-case running time for adding an element to a balanced binary search tree is O(log n), but if the tree becomes unbalanced, the worst-case running time can be O(n).

Advantages:

The average speed of inserting, retrieving, and deleting elements is quick.

Elements are still in the correct order.

Disadvantages:

The tree may become out of balance, in which case the worst-case running time may be O(n).

Compared to other data structures, trees sometimes need more memory.

Recommendation: A hash table is the suggested data structure based on the analysis. The speed of inserting and locating elements is generally fast, and the hash table's size can be changed dynamically. Hashing collisions are a possibility, but they can be reduced by using a strong hash function and employing collision-resolution techniques like chaining or open addressing. A hash table's worst-case running time for inserting n elements is O(n), which is faster than a vector's worst-case running time of O(n^22) and a tree's worst-case running time of O(n log n).