Program

Start Program

Create an empty list called courseList

Create an empty hash table called courseTable

Create an empty binary search tree called courseTree

Set dataLoaded to false

Repeat the following steps until the user chooses to exit:

Display the menu:

1. Load course data

2. Print sorted list of Computer Science courses

3. Print course title and prerequisites

9. Exit

Ask the user to enter an option

Get the user's input and save it as userOption

If userOption is 1:

Call the LoadCourseData function

Set dataLoaded to true

If userOption is 2:

If dataLoaded is false:

Show message: "You must load data first."

Else:

Call the PrintSortedCourses function

If userOption is 3:

If dataLoaded is false:

Show message: "You must load data first."

Else:

Ask the user to enter a course number

Get the user's input and save it as searchCourse

Call the PrintCourseDetails function using searchCourse

If userOption is 9:

Show message: "Exiting program..."

If userOption is anything else:

Show message: "Invalid option. Please try again."

End Repeat Loop

End Program

Print Funtion

Function PrintSortedCourses for Binary Search Tree:

Perform in-order traversal starting from the root of courseTree

During traversal:

Display each course number and name when visiting the node

End Function

| **Step** | **Time It Takes** | **How Often It Happens** | **Total Time** |
| --- | --- | --- | --- |
| **Read each line from the file** | **1 step** | **n times** | **O(n)** |
| **Split and check the line** | **1 step** | **n times** | **O(n)** |
| **Make course objects** | **1 step** | **n times** | **O(n)** |
| **Check prerequisites** | **log n steps** | **p times per course** | **O(n × p × log n)** |
| **Add to vector** | **1 step** | **n times** | **O(n)** |
| **Add to hash table** | **1 step (average)** | **n times** | **O(n)** |
| **Add to binary search tree** | **log n steps** | **n times** | **O(n log n)** |

I think the Binary Search Tree (BST) is the best choice for this program.

It automatically keeps everything in order, which makes it easier to print the course list in the right order without extra sorting. It also makes looking up specific courses pretty fast.

Even though setting it up is a bit more complex than the other options, the overall benefits make it worth it, especially since the program needs to both search for and print course information efficiently.