Alexandria Palm

CS-300

Project 1

Psuedocode

Begin Menu:

create switch cases

case 1:

call the load CSV function

case 2:

call the print function(call sort function)

case 3:

call the print course function

case 4:

Break

Begin Sort function

insertion sort (array):#takes an array as argument

check the length of the array and create a range for indexing

initialize the variable to the current index on the array

while array[index-1]>value and index>=1:

array[index]=array[index-1]

index = index - 1

array[index]=value

insertion sort (values)

order = empty string

iterate over the new list

only use positive numbers

add the numbers to the ordered list with spaces

print(order)

Complexity

Vector

O(N) Best=1, Worst=N

Hashtable

O(N^2) Best=N, Worst=2^n

Binarytree

O(Log N) Best=Log N, Worst=N Log N

The Vector is the easiet to implement in the sense that the list is formatted with simplicity and search functionality is least complex. This makes it take less time to perform the search but may take up more memory in doing larger comparisons.The Hashtable uses buckets and can save time by splitting up courses into individual strings but will require more compute power to perfom this task. This also increases the memory size to create multiple buckets of lists. The Binary tree uses recursive methods to sort and place courses by splitting the list in half as it recursively traverses thje data. This can increase memory size and runtime for large files but smooth out smaller files and result in significantly shorter runtimes.Because of these features, I would recomend the Vector. This ease of use in its search capabilities makes it the most efficient for this use case.