Shaun Ryan

Professor Andujar

CS-300: DSA: Analysis and Design

10/10/2021

**Vector Pseudocode**

**void Menu()**

INITIALIZE choice to 0

WHILE choice does not equal 4

OUTPUT menu and prompt to screen

SET choice to user input

IF choice equals 1

PROMPT for file name

SET userInput to user input

CALL LoadCourses(userInput)

ELSE IF choice equals 2

CALL PrintCourseList()

ELSE IF choice equals 3

PROMPT for course name

SET userInput to user input

CALL PrintCourseInformation(userInput)

OUTPUT “Goodbye” to the screen

**void LoadCourses()**

INITIALIZE holderString

INITIALIZE prerequisiteVector, courseVector, infileVector, courseListHolder

INITIALIZE inFileStream

OPEN dataFile

LOOP until end of datafile

SET holderString to dataFile line

INITIALIZE courseVector

LOOP until end of holderString

PUSH holderString segment to courseVector

IF index equals 0

PUSH holderString segment to courseListHolder

IF index is greater than 1

PUSH holderString segment to prerequisiteVector

IF courseVector size is less than 2

PRINT error “Not enough parameters for course”

EXIT

INITIALIZE iterator for prerequsiteVector

LOOP through prerequisiteVector

IF find(prerequisiteVector element in courseListHolder) is false

PRINT error “Prerequisite not met”

EXIT

PRINT “Course list loaded successfully”

CLOSE datafile

**void Merge(courses, i, j, k)**

SET left to i and right to j+1

WHILE left is less than or equal to j and right is less than or equal to k

IF courses[left] is less than or equal to courses[right]

ADD courses[left] to tempCourses

INCREMENT left

ELSE

ADD courses[right] to tempCourses

INCREMENT right

FOR all the remaining left side elements

ADD courses[left] to tempCourses

FOR all the remaining right side elements

ADD courses[right] to tempCourses

FOR all elements in tempCourses

SET courses equal to tempCourses

**void MergeSort(courses, i, k)**

IF i is less than k

SET j to the median of i and k

CALL MergeSort(courses, i, j)

CALL MergeSort(courses, j+1, k)

CALL Merge(courses, i, j, k)

**void PrintCourseList()**

CALL MergeSort(courses)

FOR all courses

PRINT course number

PRINT course title

FOR all prerequisites

PRINT prerequisite

Runtime Analysis

| **LoadCourses()** | **Line Cost** | **# Times Executes** | **Total Cost** |
| --- | --- | --- | --- |
| **INITIALIZE variables** | 6 | 1 | 6 |
| **OPEN dataFile** | 1 | 1 | 1 |
| **LOOP until end of dataFile** | 1 | n | n |
| **SET holderString to dataFile line** | 1 | n | n |
| **LOOP until end of holder string** | 1 | 4n | 4n |
| **PUSH segment to vector** | 1 | 4n | 4n |
| **IF index equals 0** | 1 | n | n |
| **PUSH segment to courseListHolder** | 1 | n | n |
| **IF index is greater than 1** | 1 | 2n | 2n |
| **PUSH segemet to prereqVector** | 1 | 2n | 2n |
| **IF courseVector size < 2** | 1 | n | n |
| **PRINT error message** | 1 | 0 | 0 |
| **INITIALIZE iterator** | 1 | 1 | 1 |
| **LOOP through prereqVector** | 1 | 2n | 2n |
| **COMPARE prereqVec to courseHolderList** | 1 | 2n | 2n |
| **PRINT error message** | 1 | 0 | 0 |
| **PRINT completion message** | 1 | 1 | 1 |
| **CLOSE datafile** | 1 | 1 | 1 |
| **Total Cost** | | | 21n + 10 |
| **Runtime** | | | O(n) |