David Wasson

CS 300: Project One Milestone Three: Pseudocode

Dr. Adamo

Southern New Hampshire University

6/22/2025

**Part 3- Tree Data Structure Pseudocode**

//Define structure to hold course info.

STRUCT Course

String courseNumbers

String title

String ListPrerequisites

STRUCT TreeNode

Course Numbers

TreeNode left

TreeNode right

STRUCT BinarySearchTree

TreeNode root

FUNCTION Insert(course)

FUNCTION Find(courseNumber)

FUNCTION InOrderTraversal(TreeNode root)

FUNCTION PreOrderTraversal(TreeNode root)

FUNCTION PostOrderTraversal(TreeNode root)

FUNCTION loadCourseData

OPEN at filepath

IF can’t open file

PRINT “Couldn’t open file”

RETURN

WHILE not end of the file

READ line

REMOVE any extra spaces

IF line isn’t empty

ADD line to courseLines

CLOSE file

//Gathering of courseNumbers

FOR EACH line in courseLInes

SPLIT by commas

IF less than two parts

PRINT “Missing course number or title”

RETURN

ELSE add courseNumber to courseNumbersList

FOR EACH line in courseLines

SPLIT line by commas

FOR parts second to the end

IF not in courseNumbers

PRINT “Prerequisites not found or don’t exist”

FOR EACH line in courseLInes

SPLIT lines by commas

SET courseNumber

SET title

CREATE empty list for prerequisites

FOR second part to end

ADD parts to prerequisites

CREATE Course with courseNumber, title, prerequisites

INSERT Course into courseTree

//Print all courses in alphanumeric order

FUNCTION InOrderTraversal

IF root isn’t null

InOrderTraversal(root.left)

PRINT root.course.courseNumber + “: “ + root.course.title

InOrderTraversal(root.right)

RETURN

//Search for a specific course to display title and prerequisites

FUNCTION PrintCourse by courseNumber

SET courseFound in courseTree.Find(courseNumber)

IF courseFound is null

PRINT “Course not found”

RETURN

PRINT “Course Number: “ + courseFound.courseNumber

PRINT “Title: “ + courseFound.title

IF courseFound.prerequisites is empty

PRINT “No prerequisites found”

ELSE

PRINT “Prerequisites:”

FOR EACH prerequisites in courseFound.prerequisites

PRINT “ – “ + prerequisites

Function inOrder traversal

If node isn’t null

1.First, visit all nodes in the left subtree

//inOrder(root->left)

2.Then, visit the root node

//display(root->data)

3.Visit all nodes in the right subtree

//inOrder(root->right)

Function PreOrder Traversal

If node isn’t null

1.First, visit the root node

//display(root->data)

2.Then, visit all nodes in the left subtree

//PreOrder(root->left)

3.Visit all nodes in the right subtree

//PreOrder(root->right)

Function PostOrder Traversal

If node isn’t null

1.Visit all nodes in left subtree

//PostOrder(root->left)

2.Visit all nodes in right subtree

//PostOrder(root->right)

3.Visit the root node

//display(root->data)

**Pseudocode for a menu:**

START PROGRAM

REPEAT until user selects to exit:

DISPLAY the menu:

1. Load course data from the file
2. Print all courses in alphanumeric order
3. Print course information and prerequisites for an individual course

9. Exit the program

PROMPT user for their choice

READ user choice

IF userChoice is 1:

ASK user for file name

CALL loadCourseData by file name

SET fileLoaded = true

ELSE IF userChoice is 2:

IF file hasn’t been loaded

PRINT “Please load the file first”

ELSE display courses in alphanumeric order

ELSE IF userChoice is 3:

IF file hasn’t been loaded

PRINT “Please load the file first”

ELSE ASK for course number to search

CALL print course info by courseNumber

ELSE IF userChoice is 9:

PRINT “Goodbye”

EXIT loop

ELSE:

PRINT “Choice isn’t valid. Please enter a valid choice”

END PROGRAM