3-3 Milestone: Vector Data Structure Pseudocode

Nicholle Caudy

CS300

SNHU

11/12/2023

START

//reading files

LOAD file

INITIALIZE call to open file

IF (return value is -1, the file is not found)

ELSE file is found

READ each line of file

IF (less than two values in a line)

RETURN error;

ELSE IF

Parameter has a third or more values continue first line with courseNumber (prerequisite)

ELSE

Continue to read files

CLOSE file

PRINT all course including prerequisite courseNumber at end of line

//vector data structure

DEFINE structure to hold course information

CREATE courseId variable

CREATE courseTitle variable

CREATE prerequisite1 variable

CREATE prerequisite2 variable

CREATE Linked List class

PRIVATE

Course course

Node \*next

//constructor

Node()

next = nullptr

Node (Course acourse)

course = acourse

next = nullptr

Node\* head

Node\* tail

Int size = 0

PUBLIC

virtual ~ LinkedList()

void Append(Course course)

void Prepend(Course course)

void PrintList()

Course Search (string courseId)

Int Size()

//Default constructor

SET head and tail equal to nullptr

//destructor

SET Node pointer current equal to head

Node pointer temp

WHILE

temp equals current

current equals current pointer next

delete temp

//append

CREATE new node

SET node pointer node equal new Node(course)

IF (head equals nullptr)

head = node

ELSE

If (tail does not equal nullptr)

tail -> next = node

tail = node

increase size count

//search

SET Node\* current = head

SET Node\* temp = new Node

temp -> course.courseId = “”

WHILE (current does not equal nullptr)

IF(current -> course.courseId. compare(courseId) equals 0)

Return current -> course

Current = current ->next

Return temp -> course

//print course information and prerequisites

DEFINE displayCourse (Course course)

PRINT courseId, courseTitle, prerequisite1, prerequisite2

RETURN

END