**MILESTONE 5 Pseudocode**

**Tammy Hartline**

**04/01/2023**

**CS-300 SNHU**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**IMPORT needed libraries for File PARSING, READ, WRITE, OPEN, CLOSE…**

**IN MAIN**

PROMPT user to enter *filename*

CHECK if *filename* exists and is a relevant path

IF *filename* exists and NO errors are found

OPEN *filename*

WHILE the *filename* file IS OPEN AND file IS NOT EOF (at end of file)

GET file line

IF file line has LESS than 2 values (params)

OUTPUT Error “File “ line # “is missing vital information.”

RETURN

IF file line has 2 parameters

CONTINUE to next STRUCT

IF file line has MORE than 2 parameters

CONTINUE to next STRUCT

ELSE

OUTPUT ERROR message informing user file was not opened successfully

RETURN ERROR

CLOSE file;

END

**CREATE STRUCT with Binary Search Tree for all course objects**

INITIALIZE STRUCT called Course

CREATE int courseNumber variable;

CREATE string courseName variable;

CREATE string preReqCourse variable;

DECLARE Course::CourseID();

DECLARE Course::CourseName();

DECLARE Course::CoursePreReqs();

DECLARE Course::NoPreReqCourseInfo with (CourseID, CourseName);

DECLARE Course::PreReqCourseInfo with (CourseID, CourseName, and CoursePreReqs);

WHILE NOT EOF(end of file)

CREATE delimiter(,) to separate each cell (row and column) in file.

FOR each line’s first container (content in column 1, per row),

SET container EQUAL to courseNumber;

CALL CourseID and pass in courseNumber;

FOR each line’s second container (content in column 2, per row),

SET container EQUAL to courseName;

CALL CourseName and pass in courseName;

IF line only has 2 containers (params)

CALL NoPreReqCourseInfo and PASS in

CourseID and CourseName;

ELSE

BREAK

FOR each line that contains MORE THAN 2 parameters

SET all containers that are NOT null,

AND AFTER delimiter 2 EQUAL to preReqCourse;

CALL CoursePreReqs and PASS in preReqCourse;

CALL PreReqCourseInfo and PASS in

CourseID, CourseName and CoursePreReqs;

DECLARE CourseInfoOutput function;

DECLARE SEARCH function;

IF user enters search for course number not found while reading file

OUTPUT ERROR course not found

IF user enters search for course number that has linked PreReqs

OUTPUT PreReqCourseInfo with line passed in

ELSE

OUTPUT NoPreReqCourseInfo with line passed in

CREATE PrintAllCourseInfo function

IF user selects to view all courses

OUTPUT all lines from PreReqCourseInfo & NoPreReqCourseInfo

RETURN

CLOSE FILE;

END;