Corey Gaspar

3/22/2025

**Project One Milestone One – Pseudocode**

**START**

**LOAD** libraries and CSV parser headers

**DEFINE** a struct that will hold course data

**struct Course {}**

**STRING** courseID

**STRING** courseName

**INT** preCount

**STRING** preList

**FUNCTION Course()** {

**SET** courseID to empty

**SET** courseName to empty

**SET** preCount to 0

**SET** preList to empty

}

**Main()**

**CREATE** a new list named courseList of type Course

**PRINT** “Enter CSV file path (or press enter to use default path):”

**READ** filePath

**IF** no file path is passed, use default location

**CALL** txtParser() process the CSV file and load data into the new list courseList

**CALL** validateList() to validate the data in the list courseList

**GET** user search input and **STORE** in userSearch

**CALL** printCourse() to display course information

**END**

**FUNCTION txtParser (STRING):**

**CREATE** an empty list called tempList

**OPEN** file located at either the entered file path or the default using text parsing libraries

**WHILE** not at the end of the file:

**READ** current row

**IF** row has at least two fields:

**SET** courseID to first field in row

**SET** courseName to second field in row

**SET** prerequisiteName to an empty string

**SET** preCount to 0

**WHILE** there are no more fields in row:

**INCREMENT** preCount by 1

**ADD** prerequisite to prerequisiteName, using commas for separation

**CREATE** newCourse as a Course with courseID, courseName, preCount, preNames

**ADD** newCourse to tempList

**RETURN** tempList

**END**

**FUNCTION searchList(STRING):**

**CREATE** a temporary course object (tempCourse) as a Course

**FOR EACH** course in courseList:

**IF** courseID matches the searchID:

**SET** tempCourse to course

**RETURN** tempCourse

**ELSE IF** no match found:

**RETURN** tempCourse (empty)

**END**

**FUNCTION** printCourse(STRING):

**CREATE** a temporary course object (tempCourse) as a Course

**SET** tempCourse equal to searchID

**PRINT** courseID and courseName

**FOR** i **FROM** 0 **TO** tempCourse.preCount - 1:

**PRINT** current prerequisite course from prelist

**END**

**FUNCTION validateList():**

**CREATE** tempCourse as a Course

**SET** isValid to True

**FOR** **EACH** course in courseList:

**IF** isValid is False:

**BREAK**

**FOR** i **FROM** 0 **TO** course.preCount – 1:

**SET** tempCourse to course.preList[i]

**IF** courseID is empty:

**SET** isValid to False

**BREAK**

**RETURN** isValid

**END**