**CS 300 Project One Milestone**

BEGIN

// Step 1: Define Course Structure

DEFINE STRUCT Course

courseID AS STRING

courseName AS STRING

prerequisites AS LIST OF STRING

// Step 2: Initialize Data Structure

DECLARE courseList AS VECTOR OF Course

DECLARE courseExists AS MAP OF STRING TO BOOLEAN // To validate prerequisites

// Step 3: Function to Load Courses from File

FUNCTION loadCourses(csvPath AS STRING)

DECLARE file AS fstream

OPEN csvPath FOR READING

IF file NOT OPEN

PRINT "Error: File not found"

RETURN

DECLARE line AS STRING

DECLARE tokens AS LIST OF STRING

DECLARE tempCourse AS Course

// Read Each Line Until EOF

WHILE NOT EOF(file)

READ line FROM file

SPLIT line BY ',' INTO tokens

// Validate the format

IF LENGTH(tokens) < 2

PRINT "Error: Invalid format in file"

CONTINUE

// Create course object

tempCourse.courseID = tokens[0]

tempCourse.courseName = tokens[1]

tempCourse.prerequisites = EMPTY LIST

// Store course ID for validation

courseExists[tempCourse.courseID] = TRUE

// Process prerequisites if present

FOR i FROM 2 TO LENGTH(tokens) - 1

IF tokens[i] IN courseExists

APPEND tokens[i] TO tempCourse.prerequisites

ELSE

PRINT "Error: Prerequisite " + tokens[i] + " does not exist."

CONTINUE // Skip invalid prerequisite

// Store the course object in vector

APPEND tempCourse TO courseList

CLOSE file

PRINT "Courses loaded successfully."

// Step 4: Function to Print Course Information

FUNCTION searchCourse(courseList AS VECTOR<Course>, courseID AS STRING)

FOR EACH course IN courseList

IF course.courseID == courseID

PRINT "Course ID: " + course.courseID

PRINT "Course Name: " + course.courseName

PRINT "Prerequisites: "

IF LENGTH(course.prerequisites) == 0

PRINT "None"

ELSE

FOR EACH prereq IN course.prerequisites

PRINT prereq

RETURN

PRINT "Course not found."

// Step 5: Main Execution

CALL loadCourses("courses.txt")

CALL searchCourse(courseList, "CSCI300") // Example search

END