**CS 300 Project One Milestone Two**

Student Name

Date

**CS 300 Project One Milestone Two**

**Pseudocode**

| FUNCTION HashCourseNumber(courseNumber):  COMPUTE hash value of courseNumber  RETURN hash value mod table size  FUNCTION LoadCoursesFromFile(fileName):  CREATE an empty hash table: courseTable  OPEN file with name fileName  IF file cannot be opened:  PRINT "Error: Cannot open file"  RETURN    FOR each line in file:  SPLIT line by commas into an array: courseData  IF length of courseData < 2:  PRINT "Error: Invalid line format - each line must have at least a course number and title"  CONTINUE    SET courseNumber = courseData[0]  SET courseName = courseData[1]  CREATE empty list: prerequisites    FOR i from 2 to length(courseData) - 1:  ADD courseData[i] to prerequisites  // Validate that prerequisites exist in file  FOR each prerequisite in prerequisites:  IF prerequisite not in courseTable:  PRINT "Error: Prerequisite", prerequisite, "not found in course list"  CONTINUE  // Store course in hash table  CREATE new Course object with courseNumber, courseName, prerequisites  SET hashIndex = HashCourseNumber(courseNumber)  INSERT hashIndex -> Course object into courseTable    CLOSE file  RETURN courseTable  FUNCTION PrintCourse(courseTable, courseNumber):  SET hashIndex = HashCourseNumber(courseNumber)  IF hashIndex NOT in courseTable:  PRINT "Error: Course", courseNumber, "not found"  RETURN    SET course = courseTable[hashIndex]  PRINT "Course:", course.courseNumber, "-", course.courseName  IF course.prerequisites IS EMPTY:  PRINT "Prerequisites: None"  ELSE:  PRINT "Prerequisites:", JOIN course.prerequisites with ", "  FUNCTION PrintAllCourses(courseTable):  SORT courseTable keys in ascending order  FOR each courseNumber in sorted keys:  CALL PrintCourse(courseTable, courseNumber) |
| --- |