

Task 1:

Get the list of students (IDs & grades in nested lists) who have taken a specific course.

Examples:

```
?- studentsInCourse('Robotics', Students).  
Students = [[stud06, 62], [stud09, 29]]  
  
?- studentsInCourse('Algorithms', Students).  
Students = []
```

Task 2:

Get the number of students who have taken a specific course.

Examples:

```
?- numStudents('Algorithms', Num).  
Num = 0  
  
?- numStudents('Math 1', Num).  
Num = 4
```

Task 3:

Get the maximum grade that a specific student was able to obtain.

Examples:

```
?- maxStudentGrade(stud10, MaxGrade).  
MaxGrade = 97  
  
?- maxStudentGrade(stud08, MaxGrade).  
MaxGrade = 71
```

Task 4:

Show a student's grade digits in a specific course as a list of words.

Examples:

```
?- gradeInWords(stud04, 'Database', DigitsWords).  
DigitsWords = [five, nine]  
?- gradeInWords(stud07, 'Math 2', DigitsWords).  
DigitsWords = [eight]  
?- gradeInWords(stud01, 'Programming 1', DigitsWords).  
DigitsWords = [nine, zero]
```

Task 5:

Get a list containing the sequence of courses that a student needs to take in order to be able to take the target course. The query must only succeed if the student **has already taken and successfully passed** a course that can directly or indirectly lead to the target.

Examples:

```
?- remainingCourses(stud01, 'Advanced Algorithms', Courses).  
Courses = ['OOP', 'Data Structures', 'Algorithms']  
?- remainingCourses(stud07, 'Electronics 2', Courses).  
false.  
?- remainingCourses(stud02, 'Networks', Courses).  
Courses = []  
?- remainingCourses(stud05, 'Computer Architecture', Courses).  
Courses = ['Electronics 2']  
?- remainingCourses(stud08, 'Data Warehouses', Courses).  
false.
```