CSCI 3005 – Programming Assignment 2 – Spring 2020

At a local university, caring administrators wish to ensure that classes are scheduled in such a way that students who need certain courses are not adversely affected by scheduling conflicts. Your task is to assist them in this matter by writing software that determines the number of time slots needed to schedule a set of courses, subject to the restriction that no two courses are scheduled at the same time if any of the following are true:

- They are taught by the same instructor
- They are scheduled in the same classroom
- They are in the same discipline and are both numbered 3xxx or 4xxx

As discussed, this problem can be attacked using the graph-coloring backtracking algorithm studied in class. Keep in mind that, although the scheduling restrictions will allow your program to prune the search tree significantly, for large values of n, the solution can be very computationally intensive. Your solution is to be implemented as a class named `CourseScheduler` containing the following public methods:

`CourseScheduler(String filename)`: a constructor to read in the data from a text file containing:
Header Line - A single line containing a positive integer (<= 100) representing the number of courses
Course List - A list of courses, one per line. Each line will consist of the following, separated by commas:

- *DEPT:* four-character code identifying an academic discipline
- *NUMBER:* four-digit number identifying the level of the course
- *BUILDING:* four-character code identifying building where course is taught
- *ROOM:* four-character code identifying room where course is taught
- *INSTRUCTOR:* last name, followed by a space, followed by first initial

`boolean solutionExists(int m)`: is it possible to develop a schedule with m slots?

`String getSchedule(int m)`: returns one of two results:

- The string "No solution with m slots"
- One sample schedule (resulting from a depth-first backtracking algorithm) that utilizes the specified number of time slots. For convenience, you may simply label the time slots: Time 1, Time 2, …, Time m. For the sample CBAClasses10.txt file, the resulting schedule would be:

```
Time 1:
    BUSN 1001 HANN 109 Babin L
    CINS 3044 HMPH 318 Barnett W
    RMIN 3008 HMPH 113 Berry C
    ACCT 3008 HMPH 222 Boswell K
    CONS 2011 CONS 126 Bray H
Time 2:
    MRKT 3002 HMPH 124 Babin L
    CINS 4030 HMPH 318 Barnett W
    RMIN 3020 HMPH 113 Berry C
    ACCT 4020 HMPH 222 Boswell K
Time 3:
    MRKT 3011 HMPH 124 Babin L
```

`String reachable(String from)`: returns the number and list of courses reachable from the course given as argument (in the form "DEPT ####") as produced by conducting a depth first search of the courses graph. For example, from the graph based on CBAClasses10.txt, the method call

```
reachable("BUSN 1001")
```

would return the string: `3 courses: BUSN 1001, MRKT 3002, MRKT 3011`