

Computer Science I Honors
Problem Solving: Backtracking
Assigned Tuesday 4/17, Due Thursday 4/19 11:55pm

The Problem

You are to schedule a certain number of events, given some constraints between them. The events will be numbered 1 through n . A constraint is an ordered pair of the form (i,j) which indicates that event i must occur before event j . Your goal will be to fill in the code given so that it prints out all valid permutations for ordering the schedule.

The method of solution is backtracking and basically follows the Eight Queens example shown in class very closely. You'll have to examine the code to see what everything means since I've intentionally taken out all of the comments.)

Input File Format (schedule.in)

The first line of the input file contains a single positive integer representing the number of input cases. The first line of each input case will contain two positive integers, *numEvents* (≤ 10) and *numConstraints* (≤ 20) which represent the number of events to be schedule and the number of constraints for that schedule. The next *numConstraints* lines will contain a pair of distinct integers (in between 1 and *numEvents*, inclusive), representing a constraint. The first number in the pair represents an event that must occur BEFORE the second number in the pair.

Output Format

For each output case, produce a header with the following format:

```
Schedules for case number #k:
```

where k is the one-based input case number.

Skip a blank line. Then, write out each possible schedule, one schedule per line. Each schedule should have the number of each event in the order that it is scheduled with a single space in between each number.

Skip two blank lines after the end of the output for each case.

Sample Input and Output

This is attached on the course web page next to this document.