Programming Assignment 2

CSP Solver

CSC460 Introduction to Artificial Intelligence & Machine Learning  
Endicott College • Dr. Hank Feild  
Spring 2022

| **Name(s):** |  |
| --- | --- |

# Overview

In this programming assignment, you will extend the CSP solver we worked on in class in two ways.

## Part 1: a CSP translator

First, our CSP solver requires that the user provide a CSP file, which describes the variables, domain, and constraints. This is a generic format that works with any problem that can be described as a CSP. For example, we considered files for describing the Map Coloring problem in New England and solving the 8-queens problem. Creating a CSP file can be tedious and becomes even more so as the number of variables and constraints increase. For example, the CSP file for a Sudoku puzzle is tedious to create by hand, as well as error prone.

To assist with this, you will create a program to translate a specific identification problem into a generic CSP file that our CSP solver can parse. This translator will take a more targeted, less tedious input format . For example, a Sudoku puzzle translator will take as input a file containing a table of numbers and blank spaces. You can choose from the following problems:

* Sudoku
  + input format:
    - a 9x9 comma separated values table with some cells filled out
    - see Appendix A for an example
  + constraints that must be represented in output CSP:
    - each cell must hold a number between 1–9
    - no two cells in the same row, column, or 3x3 region may share the same value
  + the variables should be named according to the row and column: spot{row}:{column}, e.g., spot1:1 is the upper left, while spot9:9 is the lower right.
* scheduler
  + inputformat:
    - first line: a comma separated list of instructors
    - second line: a comma separated list of rooms
    - third line: a comma separated list of times
    - fourth line: the number of classes that each instructor should teach
    - see Appendix A for an example
  + constraints that must be represented in output CSP:
    - the same instructor may not teach in more than one room at the same time
    - each instructor should teach exactly the number of classes specified
  + variables should be named clearly, e.g., if your variable is a combo of room and time, then {room}-{time}, e.g., JSC312-9am
* *n*-Queens
  + input format:
    - no file is needed; but the number queens (*n)* should be specified to the translator
  + constraints that must be represented in output CSP:
    - no two queens may reside in the same row, column, or diagonal of an *n* x *n* board
  + variables should be named so their meaning is clear

## Part 2: extensions to our CSP program

Second, you must completely support the following CSP algorithms. Everything should be specifiable from the CSPDriver.

* backtracking
  + filtering—only binary constraints need to be supported; ignore all constraints involving three or more variables
    - forward checking
    - constraint propagation (arc-consistency)—use the AC-3 algorithm
  + ordering
    - minimum remaining values (MRV)
    - least constraining value (LCV)
* iterative improvement
  + maximum of *n* runs
    - each run randomly sets the initial values
    - each makes up to *m* reassignments
      * if a solution isn't found, go on to the next run

Regardless of the algorithm specified by the user at run time, you should output the following in addition to the solution:

* backtracking
  + the number of states expanded
  + the maximum size of the fringe
* iterative improvement
  + the number of total runs and reassignments (summed across all runs)

You may work with up to one other person on this programming assignment.

# Learning Outcomes

By the end of this assignment, you should demonstrate a mastery of the following:

* programmatically formulate an identification problem in terms of a constraint satisfaction problem: variables, domain, and constraints
* implement the CSP algorithms we've covered in this course
* evaluate the effectiveness of a CSP algorithm
* describe the implementation differences between different CSP algorithms

# Specifications

*Before you submit, go through and check off all of the specs your submission meets.*

Your submission should meet each of the following criteria. It must pass all specifications in order to be contribute to your targeted grade bundle at the end of the semester:

* ~~program compiles and/or runs without error~~
* the source code is well organized and commented
* each source code file includes a header indicating who wrote or contributed to the code
* ~~code from any external sources, or code that relied on outside influence, includes a citation in a comment just above the corresponding code~~
* there is an implemented, working problem-specific translator class (must be one of the identification problems specified in the PA instructions)
* ~~there is an implemented, working iterative improvement class~~
* ~~the CSP class has been updated to~~ *~~correctly~~* ~~support:~~
  + ~~forward checking (for binary constraints only)~~
  + ~~constraint propagation (for binary constraints only)~~
  + ~~MRV~~
  + ~~LCV~~
* the submission includes an updated README.md file
  + it explains which identification problem translator you implemented and how to run it
  + it includes a section that states who worked on the project
  + it includes clear formatting instructions for any input files required by your translator
* the report below is completed

# Grading

As stated in the syllabus:

*Programming assignments; started in class and completed at home. These can be done individually or in pairs.*

* *Pass: the program compiles, runs mostly as expected, the output is formatted well, the source code includes a header and in-line comments, the code is well organized, and all assignment-specific criteria are met*

As also outlined in the syllabus, if your submission does not pass, you will have an opportunity to make corrections and resubmit if you choose to use a redo token.

# Report

For the more difficult version of each problem (see Appendix A; for the *n*-Queens problem, use *n=10*), complete the following tables.

| Identification problem: | (Sudoku, scheduler, *n*-Queens) |
| --- | --- |

| ***Backtracking*** | **States expanded** | **Max fringe size** |
| --- | --- | --- |
| vanilla | 1887 | 38 |
| forward checking | 1205 | 15 |
| constraint propagation (cp) | 1887 | 38 |
| cp + MRV | 1 | 1 |
| cp + MRV + LCV | 1 | 1 |

|  | **# of restarts** | **Total # of variable reassigned across restarts** | **Average # of variables reassigned per restart** |
| --- | --- | --- | --- |
| ***Incremental implementation*** |  |  |  |

Paste your the CSP generated by your translator for this problem below:

| spot1:1=9 spot1:2=7 spot1:3 spot1:4 spot1:5 spot1:6 spot1:7 spot1:8 spot1:9 spot2:1 spot2:2=6 spot2:3 spot2:4=8 spot2:5=9 spot2:6 spot2:7 spot2:8 spot2:9=4 spot3:1=8 spot3:2 spot3:3 spot3:4 spot3:5 spot3:6 spot3:7=2 spot3:8 spot3:9 spot4:1=3 spot4:2 spot4:3 spot4:4=7 spot4:5=2 spot4:6 spot4:7=5 spot4:8=6 spot4:9 spot5:1=7 spot5:2 spot5:3 spot5:4=5 spot5:5 spot5:6 spot5:7 spot5:8=4 spot5:9=2 spot6:1=4 spot6:2 spot6:3 spot6:4 spot6:5 spot6:6 spot6:7=1 spot6:8 spot6:9 spot7:1 spot7:2 spot7:3 spot7:4 spot7:5 spot7:6=8 spot7:7 spot7:8=2 spot7:9 spot8:1 spot8:2=2 spot8:3=7 spot8:4 spot8:5 spot8:6=3 spot8:7=8 spot8:8 spot8:9=1 spot9:1 spot9:2 spot9:3 spot9:4=6 spot9:5 spot9:6 spot9:7=4 spot9:8 spot9:9  1 2 3 4 5 6 7 8 9  diff spot1:2,spot1:1 spot1:3,spot1:1 spot1:3,spot1:2 spot1:4,spot1:1 spot1:4,spot1:2 spot1:4,spot1:3 spot1:5,spot1:1 spot1:5,spot1:2 spot1:5,spot1:3 spot1:5,spot1:4 spot1:6,spot1:1 spot1:6,spot1:2 spot1:6,spot1:3 spot1:6,spot1:4 spot1:6,spot1:5 spot1:7,spot1:1 spot1:7,spot1:2 spot1:7,spot1:3 spot1:7,spot1:4 spot1:7,spot1:5 spot1:7,spot1:6 spot1:8,spot1:1 spot1:8,spot1:2 spot1:8,spot1:3 spot1:8,spot1:4 spot1:8,spot1:5 spot1:8,spot1:6 spot1:8,spot1:7 spot1:9,spot1:1 spot1:9,spot1:2 spot1:9,spot1:3 spot1:9,spot1:4 spot1:9,spot1:5 spot1:9,spot1:6 spot1:9,spot1:7 spot1:9,spot1:8 spot2:1,spot1:1 spot2:2,spot1:2 spot2:2,spot2:1 spot2:3,spot1:3 spot2:3,spot2:1 spot2:3,spot2:2 spot2:4,spot1:4 spot2:4,spot2:1 spot2:4,spot2:2 spot2:4,spot2:3 spot2:5,spot1:5 spot2:5,spot2:1 spot2:5,spot2:2 spot2:5,spot2:3 spot2:5,spot2:4 spot2:6,spot1:6 spot2:6,spot2:1 spot2:6,spot2:2 spot2:6,spot2:3 spot2:6,spot2:4 spot2:6,spot2:5 spot2:7,spot1:7 spot2:7,spot2:1 spot2:7,spot2:2 spot2:7,spot2:3 spot2:7,spot2:4 spot2:7,spot2:5 spot2:7,spot2:6 spot2:8,spot1:8 spot2:8,spot2:1 spot2:8,spot2:2 spot2:8,spot2:3 spot2:8,spot2:4 spot2:8,spot2:5 spot2:8,spot2:6 spot2:8,spot2:7 spot2:9,spot1:9 spot2:9,spot2:1 spot2:9,spot2:2 spot2:9,spot2:3 spot2:9,spot2:4 spot2:9,spot2:5 spot2:9,spot2:6 spot2:9,spot2:7 spot2:9,spot2:8 spot3:1,spot1:1 spot3:1,spot2:1 spot3:2,spot1:2 spot3:2,spot2:2 spot3:2,spot3:1 spot3:3,spot1:3 spot3:3,spot2:3 spot3:3,spot3:1 spot3:3,spot3:2 spot3:4,spot1:4 spot3:4,spot2:4 spot3:4,spot3:1 spot3:4,spot3:2 spot3:4,spot3:3 spot3:5,spot1:5 spot3:5,spot2:5 spot3:5,spot3:1 spot3:5,spot3:2 spot3:5,spot3:3 spot3:5,spot3:4 spot3:6,spot1:6 spot3:6,spot2:6 spot3:6,spot3:1 spot3:6,spot3:2 spot3:6,spot3:3 spot3:6,spot3:4 spot3:6,spot3:5 spot3:7,spot1:7 spot3:7,spot2:7 spot3:7,spot3:1 spot3:7,spot3:2 spot3:7,spot3:3 spot3:7,spot3:4 spot3:7,spot3:5 spot3:7,spot3:6 spot3:8,spot1:8 spot3:8,spot2:8 spot3:8,spot3:1 spot3:8,spot3:2 spot3:8,spot3:3 spot3:8,spot3:4 spot3:8,spot3:5 spot3:8,spot3:6 spot3:8,spot3:7 spot3:9,spot1:9 spot3:9,spot2:9 spot3:9,spot3:1 spot3:9,spot3:2 spot3:9,spot3:3 spot3:9,spot3:4 spot3:9,spot3:5 spot3:9,spot3:6 spot3:9,spot3:7 spot3:9,spot3:8 spot4:1,spot1:1 spot4:1,spot2:1 spot4:1,spot3:1 spot4:2,spot1:2 spot4:2,spot2:2 spot4:2,spot3:2 spot4:2,spot4:1 spot4:3,spot1:3 spot4:3,spot2:3 spot4:3,spot3:3 spot4:3,spot4:1 spot4:3,spot4:2 spot4:4,spot1:4 spot4:4,spot2:4 spot4:4,spot3:4 spot4:4,spot4:1 spot4:4,spot4:2 spot4:4,spot4:3 spot4:5,spot1:5 spot4:5,spot2:5 spot4:5,spot3:5 spot4:5,spot4:1 spot4:5,spot4:2 spot4:5,spot4:3 spot4:5,spot4:4 spot4:6,spot1:6 spot4:6,spot2:6 spot4:6,spot3:6 spot4:6,spot4:1 spot4:6,spot4:2 spot4:6,spot4:3 spot4:6,spot4:4 spot4:6,spot4:5 spot4:7,spot1:7 spot4:7,spot2:7 spot4:7,spot3:7 spot4:7,spot4:1 spot4:7,spot4:2 spot4:7,spot4:3 spot4:7,spot4:4 spot4:7,spot4:5 spot4:7,spot4:6 spot4:8,spot1:8 spot4:8,spot2:8 spot4:8,spot3:8 spot4:8,spot4:1 spot4:8,spot4:2 spot4:8,spot4:3 spot4:8,spot4:4 spot4:8,spot4:5 spot4:8,spot4:6 spot4:8,spot4:7 spot4:9,spot1:9 spot4:9,spot2:9 spot4:9,spot3:9 spot4:9,spot4:1 spot4:9,spot4:2 spot4:9,spot4:3 spot4:9,spot4:4 spot4:9,spot4:5 spot4:9,spot4:6 spot4:9,spot4:7 spot4:9,spot4:8 spot5:1,spot1:1 spot5:1,spot2:1 spot5:1,spot3:1 spot5:1,spot4:1 spot5:2,spot1:2 spot5:2,spot2:2 spot5:2,spot3:2 spot5:2,spot4:2 spot5:2,spot5:1 spot5:3,spot1:3 spot5:3,spot2:3 spot5:3,spot3:3 spot5:3,spot4:3 spot5:3,spot5:1 spot5:3,spot5:2 spot5:4,spot1:4 spot5:4,spot2:4 spot5:4,spot3:4 spot5:4,spot4:4 spot5:4,spot5:1 spot5:4,spot5:2 spot5:4,spot5:3 spot5:5,spot1:5 spot5:5,spot2:5 spot5:5,spot3:5 spot5:5,spot4:5 spot5:5,spot5:1 spot5:5,spot5:2 spot5:5,spot5:3 spot5:5,spot5:4 spot5:6,spot1:6 spot5:6,spot2:6 spot5:6,spot3:6 spot5:6,spot4:6 spot5:6,spot5:1 spot5:6,spot5:2 spot5:6,spot5:3 spot5:6,spot5:4 spot5:6,spot5:5 spot5:7,spot1:7 spot5:7,spot2:7 spot5:7,spot3:7 spot5:7,spot4:7 spot5:7,spot5:1 spot5:7,spot5:2 spot5:7,spot5:3 spot5:7,spot5:4 spot5:7,spot5:5 spot5:7,spot5:6 spot5:8,spot1:8 spot5:8,spot2:8 spot5:8,spot3:8 spot5:8,spot4:8 spot5:8,spot5:1 spot5:8,spot5:2 spot5:8,spot5:3 spot5:8,spot5:4 spot5:8,spot5:5 spot5:8,spot5:6 spot5:8,spot5:7 spot5:9,spot1:9 spot5:9,spot2:9 spot5:9,spot3:9 spot5:9,spot4:9 spot5:9,spot5:1 spot5:9,spot5:2 spot5:9,spot5:3 spot5:9,spot5:4 spot5:9,spot5:5 spot5:9,spot5:6 spot5:9,spot5:7 spot5:9,spot5:8 spot6:1,spot1:1 spot6:1,spot2:1 spot6:1,spot3:1 spot6:1,spot4:1 spot6:1,spot5:1 spot6:2,spot1:2 spot6:2,spot2:2 spot6:2,spot3:2 spot6:2,spot4:2 spot6:2,spot5:2 spot6:2,spot6:1 spot6:3,spot1:3 spot6:3,spot2:3 spot6:3,spot3:3 spot6:3,spot4:3 spot6:3,spot5:3 spot6:3,spot6:1 spot6:3,spot6:2 spot6:4,spot1:4 spot6:4,spot2:4 spot6:4,spot3:4 spot6:4,spot4:4 spot6:4,spot5:4 spot6:4,spot6:1 spot6:4,spot6:2 spot6:4,spot6:3 spot6:5,spot1:5 spot6:5,spot2:5 spot6:5,spot3:5 spot6:5,spot4:5 spot6:5,spot5:5 spot6:5,spot6:1 spot6:5,spot6:2 spot6:5,spot6:3 spot6:5,spot6:4 spot6:6,spot1:6 spot6:6,spot2:6 spot6:6,spot3:6 spot6:6,spot4:6 spot6:6,spot5:6 spot6:6,spot6:1 spot6:6,spot6:2 spot6:6,spot6:3 spot6:6,spot6:4 spot6:6,spot6:5 spot6:7,spot1:7 spot6:7,spot2:7 spot6:7,spot3:7 spot6:7,spot4:7 spot6:7,spot5:7 spot6:7,spot6:1 spot6:7,spot6:2 spot6:7,spot6:3 spot6:7,spot6:4 spot6:7,spot6:5 spot6:7,spot6:6 spot6:8,spot1:8 spot6:8,spot2:8 spot6:8,spot3:8 spot6:8,spot4:8 spot6:8,spot5:8 spot6:8,spot6:1 spot6:8,spot6:2 spot6:8,spot6:3 spot6:8,spot6:4 spot6:8,spot6:5 spot6:8,spot6:6 spot6:8,spot6:7 spot6:9,spot1:9 spot6:9,spot2:9 spot6:9,spot3:9 spot6:9,spot4:9 spot6:9,spot5:9 spot6:9,spot6:1 spot6:9,spot6:2 spot6:9,spot6:3 spot6:9,spot6:4 spot6:9,spot6:5 spot6:9,spot6:6 spot6:9,spot6:7 spot6:9,spot6:8 spot7:1,spot1:1 spot7:1,spot2:1 spot7:1,spot3:1 spot7:1,spot4:1 spot7:1,spot5:1 spot7:1,spot6:1 spot7:2,spot1:2 spot7:2,spot2:2 spot7:2,spot3:2 spot7:2,spot4:2 spot7:2,spot5:2 spot7:2,spot6:2 spot7:2,spot7:1 spot7:3,spot1:3 spot7:3,spot2:3 spot7:3,spot3:3 spot7:3,spot4:3 spot7:3,spot5:3 spot7:3,spot6:3 spot7:3,spot7:1 spot7:3,spot7:2 spot7:4,spot1:4 spot7:4,spot2:4 spot7:4,spot3:4 spot7:4,spot4:4 spot7:4,spot5:4 spot7:4,spot6:4 spot7:4,spot7:1 spot7:4,spot7:2 spot7:4,spot7:3 spot7:5,spot1:5 spot7:5,spot2:5 spot7:5,spot3:5 spot7:5,spot4:5 spot7:5,spot5:5 spot7:5,spot6:5 spot7:5,spot7:1 spot7:5,spot7:2 spot7:5,spot7:3 spot7:5,spot7:4 spot7:6,spot1:6 spot7:6,spot2:6 spot7:6,spot3:6 spot7:6,spot4:6 spot7:6,spot5:6 spot7:6,spot6:6 spot7:6,spot7:1 spot7:6,spot7:2 spot7:6,spot7:3 spot7:6,spot7:4 spot7:6,spot7:5 spot7:7,spot1:7 spot7:7,spot2:7 spot7:7,spot3:7 spot7:7,spot4:7 spot7:7,spot5:7 spot7:7,spot6:7 spot7:7,spot7:1 spot7:7,spot7:2 spot7:7,spot7:3 spot7:7,spot7:4 spot7:7,spot7:5 spot7:7,spot7:6 spot7:8,spot1:8 spot7:8,spot2:8 spot7:8,spot3:8 spot7:8,spot4:8 spot7:8,spot5:8 spot7:8,spot6:8 spot7:8,spot7:1 spot7:8,spot7:2 spot7:8,spot7:3 spot7:8,spot7:4 spot7:8,spot7:5 spot7:8,spot7:6 spot7:8,spot7:7 spot7:9,spot1:9 spot7:9,spot2:9 spot7:9,spot3:9 spot7:9,spot4:9 spot7:9,spot5:9 spot7:9,spot6:9 spot7:9,spot7:1 spot7:9,spot7:2 spot7:9,spot7:3 spot7:9,spot7:4 spot7:9,spot7:5 spot7:9,spot7:6 spot7:9,spot7:7 spot7:9,spot7:8 spot8:1,spot1:1 spot8:1,spot2:1 spot8:1,spot3:1 spot8:1,spot4:1 spot8:1,spot5:1 spot8:1,spot6:1 spot8:1,spot7:1 spot8:2,spot1:2 spot8:2,spot2:2 spot8:2,spot3:2 spot8:2,spot4:2 spot8:2,spot5:2 spot8:2,spot6:2 spot8:2,spot7:2 spot8:2,spot8:1 spot8:3,spot1:3 spot8:3,spot2:3 spot8:3,spot3:3 spot8:3,spot4:3 spot8:3,spot5:3 spot8:3,spot6:3 spot8:3,spot7:3 spot8:3,spot8:1 spot8:3,spot8:2 spot8:4,spot1:4 spot8:4,spot2:4 spot8:4,spot3:4 spot8:4,spot4:4 spot8:4,spot5:4 spot8:4,spot6:4 spot8:4,spot7:4 spot8:4,spot8:1 spot8:4,spot8:2 spot8:4,spot8:3 spot8:5,spot1:5 spot8:5,spot2:5 spot8:5,spot3:5 spot8:5,spot4:5 spot8:5,spot5:5 spot8:5,spot6:5 spot8:5,spot7:5 spot8:5,spot8:1 spot8:5,spot8:2 spot8:5,spot8:3 spot8:5,spot8:4 spot8:6,spot1:6 spot8:6,spot2:6 spot8:6,spot3:6 spot8:6,spot4:6 spot8:6,spot5:6 spot8:6,spot6:6 spot8:6,spot7:6 spot8:6,spot8:1 spot8:6,spot8:2 spot8:6,spot8:3 spot8:6,spot8:4 spot8:6,spot8:5 spot8:7,spot1:7 spot8:7,spot2:7 spot8:7,spot3:7 spot8:7,spot4:7 spot8:7,spot5:7 spot8:7,spot6:7 spot8:7,spot7:7 spot8:7,spot8:1 spot8:7,spot8:2 spot8:7,spot8:3 spot8:7,spot8:4 spot8:7,spot8:5 spot8:7,spot8:6 spot8:8,spot1:8 spot8:8,spot2:8 spot8:8,spot3:8 spot8:8,spot4:8 spot8:8,spot5:8 spot8:8,spot6:8 spot8:8,spot7:8 spot8:8,spot8:1 spot8:8,spot8:2 spot8:8,spot8:3 spot8:8,spot8:4 spot8:8,spot8:5 spot8:8,spot8:6 spot8:8,spot8:7 spot8:9,spot1:9 spot8:9,spot2:9 spot8:9,spot3:9 spot8:9,spot4:9 spot8:9,spot5:9 spot8:9,spot6:9 spot8:9,spot7:9 spot8:9,spot8:1 spot8:9,spot8:2 spot8:9,spot8:3 spot8:9,spot8:4 spot8:9,spot8:5 spot8:9,spot8:6 spot8:9,spot8:7 spot8:9,spot8:8 spot9:1,spot1:1 spot9:1,spot2:1 spot9:1,spot3:1 spot9:1,spot4:1 spot9:1,spot5:1 spot9:1,spot6:1 spot9:1,spot7:1 spot9:1,spot8:1 spot9:2,spot1:2 spot9:2,spot2:2 spot9:2,spot3:2 spot9:2,spot4:2 spot9:2,spot5:2 spot9:2,spot6:2 spot9:2,spot7:2 spot9:2,spot8:2 spot9:2,spot9:1 spot9:3,spot1:3 spot9:3,spot2:3 spot9:3,spot3:3 spot9:3,spot4:3 spot9:3,spot5:3 spot9:3,spot6:3 spot9:3,spot7:3 spot9:3,spot8:3 spot9:3,spot9:1 spot9:3,spot9:2 spot9:4,spot1:4 spot9:4,spot2:4 spot9:4,spot3:4 spot9:4,spot4:4 spot9:4,spot5:4 spot9:4,spot6:4 spot9:4,spot7:4 spot9:4,spot8:4 spot9:4,spot9:1 spot9:4,spot9:2 spot9:4,spot9:3 spot9:5,spot1:5 spot9:5,spot2:5 spot9:5,spot3:5 spot9:5,spot4:5 spot9:5,spot5:5 spot9:5,spot6:5 spot9:5,spot7:5 spot9:5,spot8:5 spot9:5,spot9:1 spot9:5,spot9:2 spot9:5,spot9:3 spot9:5,spot9:4 spot9:6,spot1:6 spot9:6,spot2:6 spot9:6,spot3:6 spot9:6,spot4:6 spot9:6,spot5:6 spot9:6,spot6:6 spot9:6,spot7:6 spot9:6,spot8:6 spot9:6,spot9:1 spot9:6,spot9:2 spot9:6,spot9:3 spot9:6,spot9:4 spot9:6,spot9:5 spot9:7,spot1:7 spot9:7,spot2:7 spot9:7,spot3:7 spot9:7,spot4:7 spot9:7,spot5:7 spot9:7,spot6:7 spot9:7,spot7:7 spot9:7,spot8:7 spot9:7,spot9:1 spot9:7,spot9:2 spot9:7,spot9:3 spot9:7,spot9:4 spot9:7,spot9:5 spot9:7,spot9:6 spot9:8,spot1:8 spot9:8,spot2:8 spot9:8,spot3:8 spot9:8,spot4:8 spot9:8,spot5:8 spot9:8,spot6:8 spot9:8,spot7:8 spot9:8,spot8:8 spot9:8,spot9:1 spot9:8,spot9:2 spot9:8,spot9:3 spot9:8,spot9:4 spot9:8,spot9:5 spot9:8,spot9:6 spot9:8,spot9:7 spot9:9,spot1:9 spot9:9,spot2:9 spot9:9,spot3:9 spot9:9,spot4:9 spot9:9,spot5:9 spot9:9,spot6:9 spot9:9,spot7:9 spot9:9,spot8:9 spot9:9,spot9:1 spot9:9,spot9:2 spot9:9,spot9:3 spot9:9,spot9:4 spot9:9,spot9:5 spot9:9,spot9:6 spot9:9,spot9:7 spot9:9,spot9:8 |
| --- |

# Appendix A

Example input files for certain CSP translators. See the CSP files in the csp/ folder of the class code base to see what the CSP files look like for these examples.

## Sudoku

Example 1 (corresponds to csp/sudoku1.csp in the codebase):

| 5,3,0,0,7,0,0,0,0  6,0,0,1,9,5,0,0,0  0,9,8,0,0,0,0,6,0  8,0,0,0,6,0,0,0,3  4,0,0,8,0,3,0,0,1  7,0,0,0,2,0,0,0,6  0,6,0,0,0,0,2,8,0  0,0,0,4,1,9,0,0,5  0,0,0,0,8,0,0,7,9 |
| --- |

Example 2 (this is a much more difficult puzzle):

| 9,7,0,0,0,0,0,0,0  0,6,0,8,9,0,0,0,4  8,0,0,0,0,0,2,0,0  3,0,0,7,2,0,5,6,0  7,0,0,5,0,0,0,4,2  4,0,0,0,0,0,1,0,0  0,0,0,0,0,8,0,2,0  0,2,7,0,0,3,8,0,1  0,0,0,6,0,0,4,0,0 |
| --- |

## Scheduler

Example 1 (corresponds to csp/schedule1.csp and csp/schedule2.csp) in the codebase):

| Diehl,Feild,Ocean  JSC212,JSC312  9am,10am,11am  2 |
| --- |

Example 2 (a much harder problem—this will take longer to run):

| Caterina,Diehl,Feild,,Key,Lombardo,McDaniel,Ocean  JSC212,JSC312,JSC230,JSC128  9am,10am,11am,12pm,1pm,2pm,3pm  4 |
| --- |