Artificial Intelligence 1
Assignment6: Constraint Propagation
– Given Dec. 1, Due Dec. 11 –

Problem 6.1 (Scheduling CS Classes as a CSP) 40 pt

You are in charge of scheduling for computer science classes. There are 5 classes and 3 professors to teach them. You are constrained by the fact that each professor can only teach one class at a time. The classes are:

- Class 1 - *Intro to Artificial Intelligence*: meets 8:30-9:30am,
- Class 2 - *Intro to Programming*: meets 8:00-9:00am,
- Class 3 - *Natural Language Processing*: meets 9:00-10:00am,
- Class 4 - *Machine Learning*: meets 9:30-10:30am,
- Class 5 - *Computer Vision*: meets 9:00-10:00am.

The professors are:

- Professor A, who is available to teach Classes 1, 2, 3, 4, 5.
- Professor B, who is available to teach Classes 3 and 4.
- Professor C, who is available to teach Classes 2, 3, 4, and 5.

1. Formulate this problem as a binary CSP problem in which there is one variable per class, stating the domains, and constraints. Constraints should be specified formally and precisely.
2. Give the constraint graph associated with your CSP.
3. Give examples of
   - a total inconsistent assignment
   - a solution

Problem 6.2 (Scheduling CS Classes with Constraint Propagation) 30 pt

Consider the CSP problem for scheduling CS classes from the previous assignment.

1. Show the CSP obtained by running arc-consistency. As usual, you can visualize this as a graph whose
   - nodes are labeled with the variable names and domains
   - edges are labeled with the constraints.
2. Give all optimal cutsets for the CSP.

Problem 6.3 (CSP Formalization) 30 pt

Consider the following binary CSP $\Pi := (V, D, C)$:

- Variables $V = \{x, y, z\}$
- Domains D: $D_x = \{0, 1, 2\}$, $D_y = \{1, 2\}$, $D_z = \{0, 1\}$
- Constraints C: $x \neq y$, $y > z$

1. Give all pairs (v, w) of variables such that v is arc-consistent relative to w.
2. Give all solutions that would remain if we added the constraint $x \neq z$ to Π.
3. Assume we assign $y = 1$ in Π and apply forward-checking. Give the resulting domains D_x, D_y, D_z.

Problem 6.4 (Kalah Tournament) 200 pt

This is an extraordinary problem, in which we implement adversarial search as a tournament. You can implement all search methods, e.g., to simulate a move or compute the full game tree etc.

Submission parameters:

- Team size: 3 people per team
- Deadline: 2023-01-08
- Site: The submission site will be opened later.
- Format: The details will be published later on the studon forum. But you can use any programming language, and your program might be subject to resource constraints (overall space for the binary, time per move, etc.).
- Points: Submissions that are better than a relatively low baseline (e.g., win against a player that makes random moves) will receive 100 points. The team with the best agent receives an additional 100 points, the 2nd team 90 points, the 3rd 80 etc.

Further details will be announced in the forum as they come up.