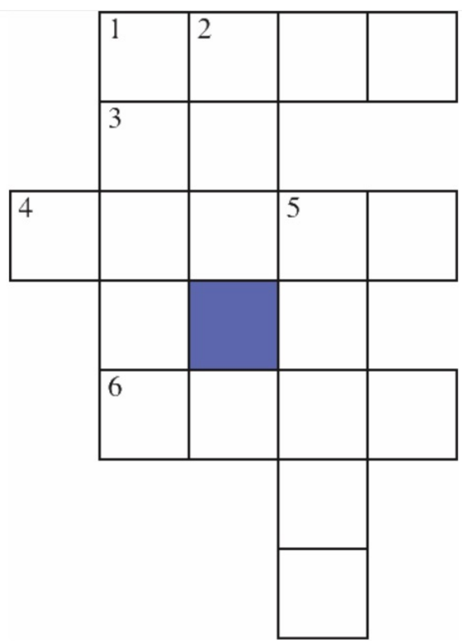
ICS 271
Fall 2017
Instructor : Kalev Kask
Homework Assignment 4
Due Tuesday 11/7

1. (15 points) Consider the crossword puzzle



We represent the problem as a CSP where there is a variable for each of the positions where a word is supposed to go (1-across, 2-down, 3-across, 4-across, 5-down, 6-across). Domains of variables are words from this list : at, eta, be, hat, he, her, it, him, on, one, desk, dance, usage, easy, dove, first, else, loses, fuels, help, haste, given, kind, sense, soon, sound, this, think. The constraints are that the letter is the same where the words intersect.

Simulate execution of Stochastic Local Search on this problem. Start with v1=desk, v2=eta, v3=at, v4=dance, v5=dance, v6=desk. Define a cost function for this problem. Show the first 3 iterations of SLS. For each iteration, show the current cost, and the cost of each node in the neighborhood, as well as the new assignment chosen.

2. (10 points) Give a precise formulation of the following constraint satisfaction problem (i.e. define variables, domains of variables, constraints):

   (a) Class scheduling: There is a fixed number of professors and classrooms, a list of classes to be offered, and a list of possible time slots for classes. Each professor has a set of classes that he or she can teach.

3. (15 points) Consider the following binary constraint network: There are 4 variables: X_1, X_2, X_3, X_4, with the domains:

   $D_1 = \{1, 2, 3, 4\}$, $D_2 = \{3, 4, 5, 8, 9\}$, $D_3 = \{2, 3, 5, 6, 7, 9\}$, $D_4 = \{3, 5, 7, 8, 9\}$.

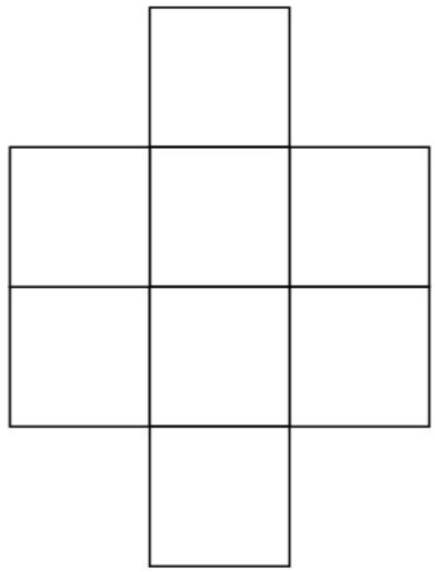
   The 3 constraints are: $X_1 \geq X_2$, $X_2 > X_3$ or $X_3 - X_2 = 2$, $X_3 \neq X_4$.

   (a) Write the constraints in a relational form and draw the constraint graph.

   (b) Is the network arc-consistent? if not, compute the arc-consistent network.

   (c) Is the network consistent (i.e. does it have a solution)? If yes, give a solution.

4. (15 points) Consider the 8 squares positioned as follows:



The task is to label the squares above with the numbers 1-8 such that the labels of any pair of adjacent squares (i.e. horizontal, vertical or diagonal) differ by at least 2. Assume that there is a variable for each square in the grid with a domain of {1,...,8}.

(a) Write the constraints in a relational form and draw the constraint graph.

(b) Is the network arc-consistent? if not, compute the arc-consistent network.

(c) Is the network consistent (i.e. does it have a solution)? If yes, give a solution.

5. (20 points) Consider a constraint satisfaction problem with 15 variables $X_1, \ldots, X_{15}$, and domains of variables $D_1, \ldots, D_{15} = \{1, 2, 3, 4, 5, 6, 7, 9\}$, *the constraint graph is a perfect binary tree with degree 2*, and the binary constraints C_{ij} are $X_i > X_j + 1$ where i is the parent of j in the tree (the root is X_1, see Figure 1).

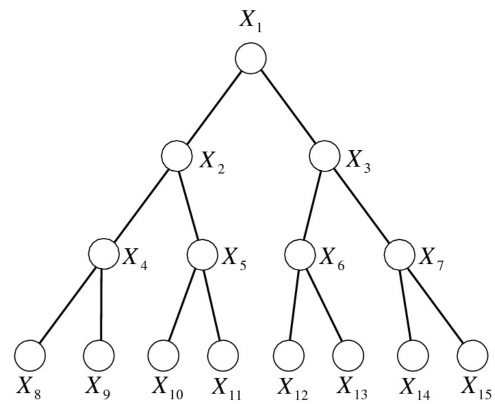


Figure 1: A constraint tree problem

(a) Is the network arc-consistent? if not, enforce arc-consistency.

(b) Is the network consistent? If yes, compute a solution.

(c) Can you suggest a variable ordering for which backtrack search will be backtrack-free, assuming the problem is arc-consistent?

(d) Bound the complexity of solving any problem whose constraint graph is the same as for this problem.

6. (15 points) (Problem 6.5 in Russel and Norvig) Solve the cryptarithmetic problem in Figure 6.2 by hand, using backtracking with MRV variable order heuristic (use Forward Checking) and with least constraining value heuristic.

7. (15 points) Consider the following algorithms

- arc consistency with domain splitting. This algorithm (recursively, until each variable has domain of size at most 1) splits the problem into disjoint smaller subproblems by splitting the domain of some variable; then each subproblem is solved separately, and solutions to each subproblem are combined to form a solution to the original problem. Before each domain splitting, we enfore arc consistency.
- variable elimination. This algorithm (recursively, until one variable remains) picks a variable to eliminate, combines all constraints involving that variable, and projects the resulting combined constraint onto remaining variables.

- stochastic local search.
- backtracking search.

Which of these algorithms can

(a) determine that there is no solution, if the problem is inconsistent?

(b) find a solution if one exists?

(c) guarantee to find all solutions?