Exercises Lecture 9
Intelligent Systems Programming (ISP)

Exercise 1 (adapted from Mozart-OZ online tutorial)

A CSP model consists of a set of variables, a set of domains (possible values) for each variable and a set of constraints between the variables that limit the values that variables can take, all describing a specific problem.

The code of Professor Smart's safe is a sequence of 4 distinct (i.e., all different) nonzero digits, $C_1, \ldots, C_4$ such that the following equations and inequalities are satisfied:

$C_1 \neq 1$, $C_2 \neq 2$, $C_3 \neq 3$, $C_4 \neq 4$

$C_4 - C_3 > 2$

$C_3 + C_1 \geq 5$

$C_1 * C_2 \geq 6$

$C_1 + C_2 \leq 7$

$C_3 \leq 5$

$C_4 > 4$

A CSP model for this problem is:

Variables: $\{C_1, C_2, C_3, C_4\}$

Domains: $\{[1\ldots6], [1\ldots6], [1\ldots6], [1\ldots6]\}$

Constraints: $\{C_1 \neq C_2, C_1 \neq C_3, C_1 \neq C_4, C_2 \neq C_3, C_2 \neq C_4, C_3 \neq C_4, C_1 \neq 1, C_2 \neq 2, C_3 \neq 3, C_4 \neq 4, C_4 - C_3 > 2, C_3 + C_1 \geq 5, C_1 * C_2 \geq 6, C_1 + C_2 \leq 7, C_3 \leq 5, C_4 > 4\}$

Now what you have to do:

a) Use backtracking to find a solution, assume the order $\{C_1, C_2, C_3, C_4\}$ for the selection of the variables, draw the search tree and show in the failed leaves which constraint is the one that fails.
b) Use FC and MRV to find a solution, draw the search tree and show the new domains of the variables for each leave after the execution of FC. Make the CSP node consistent before you start the search.

Exercise 2 (adapted from Mozart-OZ online tutorial)

Betty, Chris, Donald, Fred, Gary, Mary, and Paul want to align in one row for taking a photo. Some of them have preferences next to whom they want to stand:

1. Betty wants to stand next to Gary and Mary.
2. Chris wants to stand next to Gary.
3. Fred wants to stand next to Donald.
4. Paul wants to stand next to Donald.

Be aware that 2 people can't stand in the same position at the same time, meaning that there is a different constraint between each pair of variables representing the position where two people stand.

We can define such a constraint using mathematical operators: *diff*(X, Y): $X \neq Y$, where X and Y are variables.

A partial CSP model for this problem is:

Variables: {B, C, D, F, G, M, P}

Domains: {[$1 \ldots 7$], [$1 \ldots 7$], [$1 \ldots 7$], [$1 \ldots 7$], [$1 \ldots 7$], [$1 \ldots 7$], [$1 \ldots 7$]}

Constraints: {$B \neq C$, $B \neq D$, $B \neq F$, $B \neq G$, $B \neq M$, $B \neq P$, $C \neq D$, $C \neq F$, $C \neq G$, $C \neq M$, $C \neq P$, $D \neq F$, $D \neq G$, $D \neq M$, $D \neq P$, $F \neq G$, $F \neq M$, $F \neq P$, $M \neq G$, $M \neq P$, $G \neq P$}

a) Define a constraint *adj*(X, Y) representing the fact that for a possible value of X, the value of Y has to be next to it. Use logical and mathematical operators to do so.

b) As you may remember from the lecture, a constraint graph is a graph where each node represents a variable and each edge represents one of the binary constraints defined in the set of constraints. Draw a constraint graph for a CSP' just considering the adjacency constraints of this problem. Variables and domains are the same as defined above.

c) Add the adjacency constraints together with constraint $B=1$ to the CSP and run the **AC-3** algorithm in order to make it arc-consistent. Show the domains of the variables after the execution of the algorithm.

Exercise 3

Johnny Seafood works at the aquarium in his town. He is attending some courses at the local university and now he has a problem at work where he might be able to apply what he has learned in school. The aquarium where Johnny works has recently got 5 new fish species that need to be allocated in the three available tanks. There are some constraints concerning what species can be together in a tank and also some specific requirements.

- Sharks and dolphins can't be together in the same tank.
- Sharks and remoras must be in the same tank.
- Dolphins can't be in tank 3. It is not possible for them to perform their show in this tank.

- There are too many dolphins so they cannot share the tank with the whale.
- The tank closest to the food deposit is tank 2. Johnny wants to allocate the whale there so he doesn't have to carry its food too far.
- Remoras are so small that the whale could eat them. Thus, they can't be in the same tank as the whale.
- Lobsters can be anywhere.

a) Define the problem as a CSP. That is, define the variables, domains, and constraints of the problem. All your constraints should be binary.
b) Draw the constraint graph of the CSP.
c) Use the Forward Checking algorithm to solve the problem, make the CSP node consistent before starting the algorithm. Draw the search tree and the domain of the variables at each node. Assume the fixed variable order: sharks, lobsters, remoras, dolphins, and whale.
d) Use the MAC algorithm to solve the problem. Draw the search tree and the domain of the variables at each node. Assume the same order for selecting the variables as in c).

Mandatory assignment

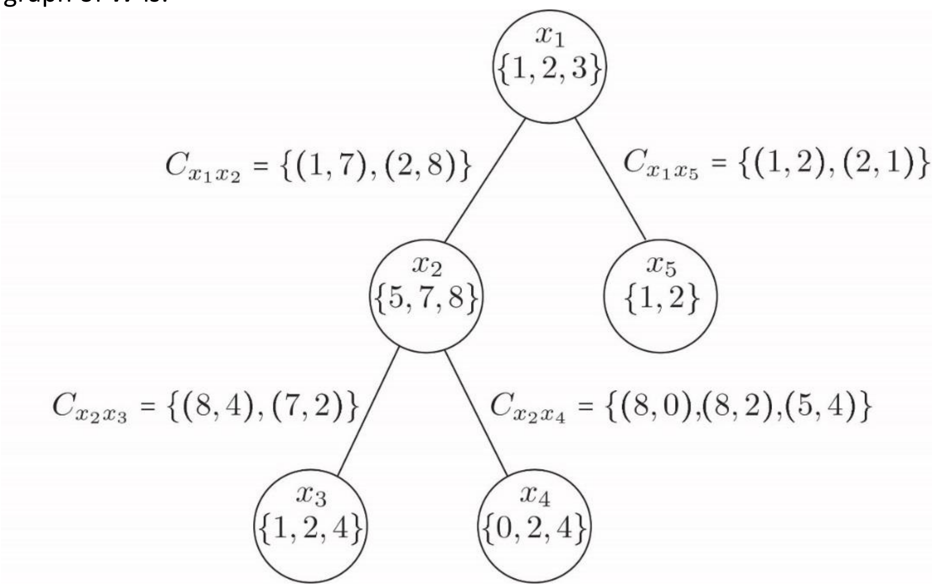
In this assignment, we consider binary CSPs where the constraint graph forms a tree. Recall that the nodes and edges of the constraint graph represent variables and constraints, respectively. As an example consider the binary CSP $W = (X,D,C)$ where:

$X = \{x_1, x_2, x_3, x_4, x_5\}$;

$D_1 = \{1,2,3\}$, $D_2 = \{5,7,8\}$, $D_3 = \{1,2,4\}$, $D_4 = \{0,2,4\}$, $D_5 = \{1,2\}$;

$C = \{$ $Cx_1x_2=\{(1,7),(2,8)\}$,
$Cx_1x_5=\{(1,2),(2,1)\}$,
$Cx_2x_3=\{(8,4),(7,2)\}$,
$Cx_2x_4=\{(8,0),(8,2),(5,4)\}$ $\}$.

The constraint graph of W is:



1) Reduce the domains of the variables of *W* such that *W* becomes arc consistent.
2) Write a solution to *W*.
3) Describe in words a polynomial time algorithm that can find a solution to an arbitrary binary CSP where the constraint graph forms a tree.
   (Hint: use REVISE on the arcs in the tree, eg. from parent to child, and/or from child to parent.)