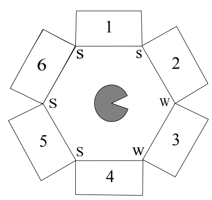
CS188 Fall 2017 Section 3: CSPs + Games

1 CSPs: Trapped Pacman

Pacman is trapped! He is surrounded by mysterious corridors, each of which leads to either a pit (P), a ghost (G), or an exit (E). In order to escape, he needs to figure out which corridors, if any, lead to an exit and freedom, rather than the certain doom of a pit or a ghost.

The one sign of what lies behind the corridors is the wind: a pit produces a strong breeze (S) and an exit produces a weak breeze (W), while a ghost doesn't produce any breeze at all. Unfortunately, Pacman cannot measure the strength of the breeze at a specific corridor. Instead, he can stand *between* two adjacent corridors and feel the max of the two breezes. For example, if he stands between a pit and an exit he will sense a strong (S) breeze, while if he stands between an exit and a ghost, he will sense a weak (W) breeze. The measurements for all intersections are shown in the figure below.

Also, while the total number of exits might be zero, one, or more, Pacman knows that two neighboring squares will *not* both be exits.



Pacman models this problem using variables X_i for each corridor i and domains P, G, and E.

1. State the binary and/or unary constraints for this CSP (either implicitly or explicitly).

   Binary:

   $X_1 = P$ or $X_2 = P$, $X_2 = E$ or $X_3 = E$,
   $X_3 = E$ or $X_4 = E$, $X_4 = P$ or $X_5 = P$,
   $X_5 = P$ or $X_6 = P$, $X_1 = P$ or $X_6 = P$,
   $\forall i, j$ s.t. $\text{Adj}(i, j)$ $\neg(X_i = E$ and $X_j = E)$

   Unary:

   $X_2 \neq P$,
   $X_3 \neq P$,
   $X_4 \neq P$

   Note: This is just one of many solutions. The answers below will be based on this formulation.

2. Suppose we assign X_1 to E. Perform forward checking after this assignment. Also, enforce unary constraints.

X_1			E
X_2			
X_3		G	E
X_4		G	E
X_5	P	G	E
X_6	P		

3. Start over from the initial configuration after unary constraints are enforced. Cross out the values from the domains of the variables that will be deleted by enforcing arc consistency.

X_1	P		
X_2		G	E
X_3		G	E
X_4		G	E
X_5	P		
X_6	P	G	E

4. According to MRV, which variable or variables could the solver assign first?

   X_1 or X_5 (tie breaking)

5. Assume that Pacman knows that $X_6 = G$. List all the solutions of this CSP or write *none* if no solutions exist.

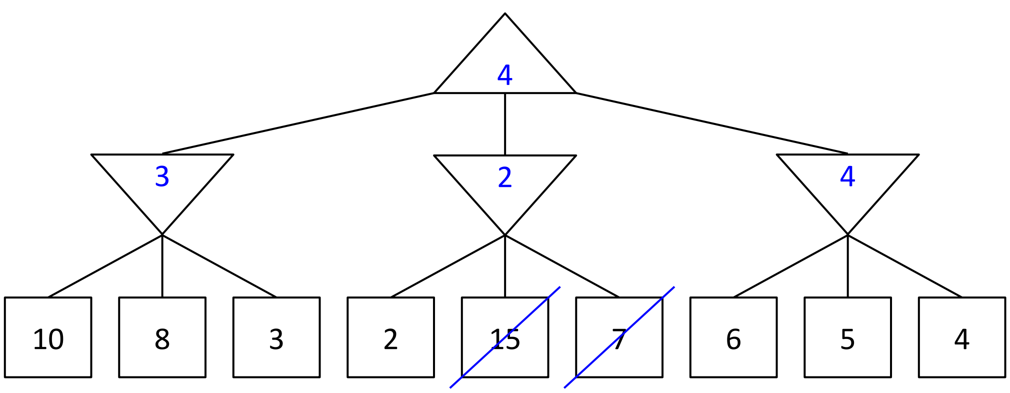
   (P,E,G,E,P,G)
   (P,G,E,G,P,G)

6. If standard backtracking search were run on a circle-structured graph, enforcing arc consistency at every step, what, if anything, can be said about the worst-case backtracking behavior (e.g. number of times the search could backtrack)?

   A tree structured CSP can be solved without any backtracking. Thus, the above circle-structured CSP can be solved after backtracking at most d times, since we might have to try up to d values for X_j before finding a solution.

2 Games

1. Consider the zero-sum game tree shown below. Triangles that point up, such as at the top node (root), represent choices for the maximizing player; triangles that point down represent choices for the minimizing player. Assuming both players act optimally, fill in the minimax value of each node.



2. Which nodes can be pruned from the game tree above through alpha-beta pruning? If no nodes can be pruned, explain why not. Assume the search goes from left to right; when choosing which child to visit first, choose the left-most unvisited child.