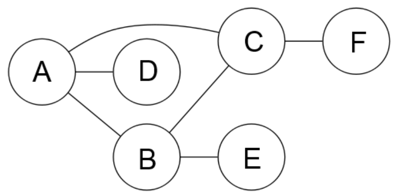
CS188: Exam Practice Session 2

Q1. CSPs

(a) The graph below is a constraint graph for a CSP that has only binary constraints. Initially, no variables have been assigned.

For each of the following scenarios, mark all variables for which the specified filtering might result in their domain being changed.



(i) A value is assigned to A. Which domains might be changed as a result of running forward checking for A?

○ A ○ B ○ C ○ D ○ E ○ F

(ii) A value is assigned to A, and then forward checking is run for A. Then a value is assigned to B. Which domains might be changed as a result of running forward checking for B?

○ A ○ B ○ C ○ D ○ E ○ F

(iii) A value is assigned to A. Which domains might be changed as a result of enforcing arc consistency after this assignment to A?

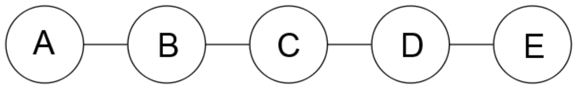
○ A ○ B ○ C ○ D ○ E ○ F

(iv) A value is assigned to A, and then arc consistency is enforced. Then a value is assigned to B. Which domains might be changed as a result of enforcing arc consistency after the assignment to B?

○ A ○ B ○ C ○ D ○ E ○ F

(b) You decide to try a new approach to using arc consistency in which you initially enforce arc consistency, and then enforce arc consistency every time you have assigned an even number of variables.

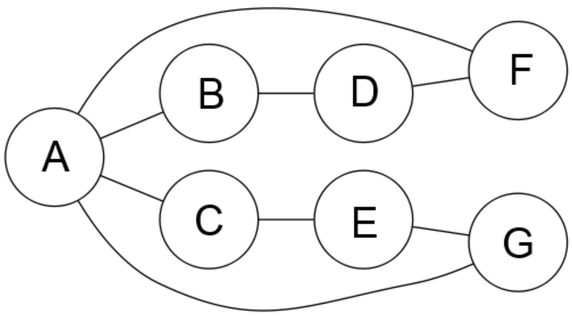
You have to backtrack if, after a value has been assigned to a variable, X, the recursion returns at X without a solution. Concretely, this means that for a single variable with d values remaining, it is possible to backtrack up to d times. For each of the following constraint graphs, if each variable has a domain of size d, how many times would you have to backtrack in the worst case for each of the specified orderings?

(i) 

A-B-C-D-E: ____________________

A-E-B-D-C: ____________________

C-B-D-E-A: ____________________

(ii) 

A-B-C-D-E-F-G: ____________________

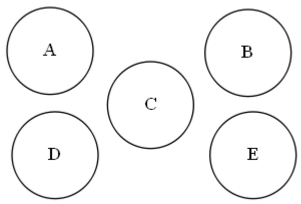
F-D-B-A-C-G-E: ____________________

C-A-F-E-B-G-D: ____________________

(c) Consider a modified CSP in which we wish to find *every possible satisfying assignment*, rather than just one such assignment as in normal CSPs. In order to solve this new problem, consider a new algorithm which is the same as the normal backtracking search algorithm, except that when it sees a solution, instead of returning it, the solution gets added to a list, and the algorithm backtracks. Once there are no variables remaining to backtrack on, the algorithm returns the list of solutions it has found.

For each graph below, select whether or not using the MRV and/or LCV heuristics could affect the number of nodes expanded in the search tree in this new situation.

(i)



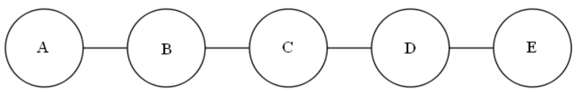
○ Neither MRV nor LCV can have an effect.

○ Only MRV can have an effect.

○ Only LCV can have an effect .

○ Both MRV and LCV can have an effect.

(ii)



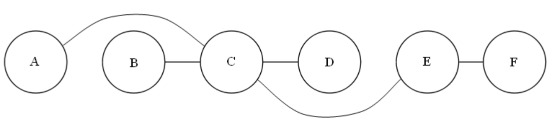
○ Neither MRV nor LCV can have an effect.

○ Only MRV can have an effect.

○ Only LCV can have an effect .

○ Both MRV and LCV can have an effect.

(iii)



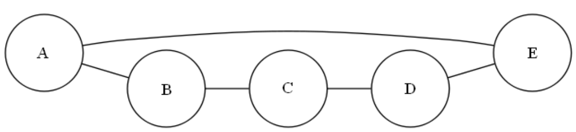
○ Neither MRV nor LCV can have an effect.

○ Only MRV can have an effect.

○ Only LCV can have an effect .

○ Both MRV and LCV can have an effect.

(iv)



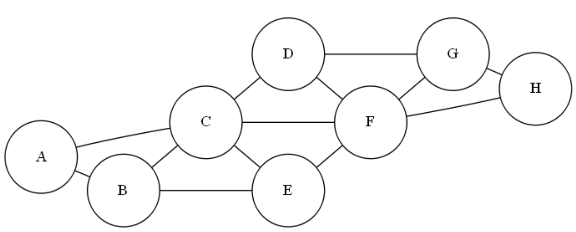
○ Neither MRV nor LCV can have an effect.

○ Only MRV can have an effect.

○ Only LCV can have an effect .

○ Both MRV and LCV can have an effect.

(v)



○ Neither MRV nor LCV can have an effect.

○ Only MRV can have an effect.

○ Only LCV can have an effect .

○ Both MRV and LCV can have an effect.