CS 188 Fall 2021 Introduction to Artificial Intelligence Challenge Q7 HW2

- **Due:** Tuesday 9/14 at 10:59pm.
- **Policy:** Can be solved in groups (acknowledge collaborators) but must be submitted individually.
- **Make sure to show all your work and justify your answers**.
- **Note:** This is a typical exam-level question. On the exam, you would be under time pressure, and have to complete this question on your own. We strongly encourage you to first try this on your own to help you understand where you currently stand. Then feel free to have some discussion about the question with other students and/or staff, before independently writing up your solution.
- Your submission on Gradescope should be a PDF that matches this template. Each page of the PDF should align with the corresponding page of the template (page 1 has name/collaborators, question begins on page 2.). **Do not reorder, split, combine, or add extra pages**. The intention is that you print out the template, write on the page in pen/pencil, and then scan or take pictures of the pages to make your submission. You may also fill out this template digitally (e.g. using a tablet.)

First name	NAMEERA
Last name	FAISAL AKHTAR
SID	3034244256
Collaborators	SARA IMAM

For staff use only:

Q7.	Challenge Problem (CSP)	/18

Q7. [18 pts] Challenge Problem (CSP)

Pacman has reconciled with 4 of the ghosts Bubbly, Kozy, Irresistible and Cuddly and he wants to invite them to live with him and Ms. Pacman in their 6 bedroom house. To figure out who sleeps in which bedroom, Pacman wants to solve a CSP with the variables being Pacman (P), Ms. Pacman (M), Bubbly (B), Kozy (K), Irresistible (I) and Cuddly (C). The values are the different room numbers, from 1 to 6 and the constraints that must be satisfied are as follows:

A. Only one person can take up a room.

B. $P > 3$ Unary

C. $K < P$

D. M is either 5 or 6 Unary

E. $P > M$

F. B is even Unary

G. I is not 1 or 6 Unary

H. $|I - C| = 1$

I. $|P - B| = 2$

7.1) (2pts) If we enforce unary constraints, which of the following values will be eliminated from each domain? Cross them out.

P	~~1~~	~~2~~	~~3~~	4	5	6
B	~~1~~	2	~~3~~	4	~~5~~	6
C	1	2	3	4	5	6
K	1	2	3	4	5	6
I	~~1~~	2	3	4	5	~~6~~
M	~~1~~	~~2~~	~~3~~	~~4~~	5	6

B) P > 3. Thus, the domain for P only has 4, 5, 6 now.

D) M is either 5 or 6. All other values to be crossed out.

F) B is even. Thus, the domain for B only has 2, 4, 6 now.

G) I is not 1 or 6. Thus, the domain for I has 2, 3, 4, 5 now.

7.2) (1pts) If we follow the Minimum Remaining Value (MRV) method, which variable will be assigned first?

MRV chooses M as MRV chooses the variable w/ the smallest domain (in this case, 2).

7.3) (2pts) Now let's assume that we choose to assign P first with the value of 6. If we enforce unary constraints (from 7.1)) and run forward checking, what will the new domains look like? Cross out the values that will be removed.

P						6
B	~~1~~	~~2~~	~~3~~	4	~~5~~	~~6~~
C	1	2	3	4	5	~~6~~
K	1	2	3	4	5	~~6~~
I	~~1~~	2	3	4	5	~~6~~
M	~~1~~	~~2~~	~~3~~	~~4~~	5	~~6~~

A) Remove all 6s.

B) No change

C) No change.

D) No change

E) No change.

F) No change

G) No change

H) No change

I) B: Cross out 2

7.4) (3pts) We no longer want to run backtracking search, and so we decide to start over and run iterative improvement with min-conflicts heuristic for value selection. Let's assume that the starting assignment is P:6, M:5, B:4, C:3, K:2, I:1.

First fill out the left table with the number of constraint violations per variable. Next, in the table on the right, put for each variable an X in the box corresponding to each value that that variable could take following the min-conflicts method. Do not mark the current value of each variable with an X.

Variable	# violated
P	0
B	0
C	1
K	0
I	2
M	0

	1	2	3	4	5	6
P						
B						
C		X				
K						
I		X		X		
M						

A. Only one person can take up a room.

B. $P > 3$

C. $K < P$

D. M is either 5 or 6

E. $P > M$

F. B is even

G. I is not 1 or 6 I violates

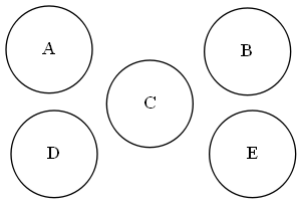
H. $|I - C| = 1$ I, C violates

I. $|P - B| = 2$

Now let's consider a different problem. In our usual CSPs our goal is to find an constraint satisfying assignment. For this problem our task is to try and find all possible such assignments. This new CSP solving algorithm is similar to the backtracking algorithm we saw in class with the only difference being that once it finds a solution it appends it to a list instead of returning it. After that the algorithm backtracks and once it cannot backtrack anymore it returns the aforementioned list.

In the following questions determine whether using the MRV and the LCV methods will cause any change on the number of nodes that will be expanded after using our modified algorithm.

7.5) (2pts)



(A.) No change from MRV and LCV

B. Change only from MRV

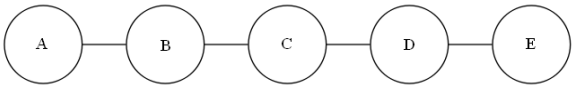
C. Change only from LCV

D. Change from both

Since all the variables are disjoint, only unary constraints will affect this CSP.

The variables do not affect eachother, so MRV and LCV will not cause a change.

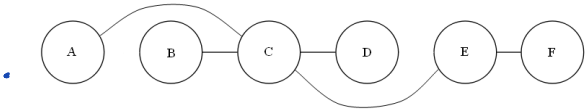
7.6) (2pts)



A. No change from MRV and LCV
B. Change only from MRV
C. Change only from LCV
D. Change from both

MRV is a variable ordering scheme while LCV is a value ordering scheme. As a result, LCV does not change the number of nodes expanded because since our algorithm finds ALL possible assignments, we need to check the value for each variable anyway, so adding LCV will have no benefit or detriment. On the other hand, MRV could change the number of nodes expanded since it changes the order of the search.

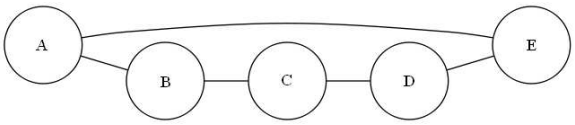
7.7) (2pts)



A. No change from MRV and LCV
B. Change only from MRV
C. Change only from LCV
D. Change from both

MRV is a variable ordering scheme while LCV is a value ordering scheme. As a result, LCV does not change the number of nodes expanded because since our algorithm finds ALL possible assignments, we need to check the value for each variable anyway, so adding LCV will have no benefit or detriment. On the other hand, MRV could change the number of nodes expanded since it changes the order of the search.

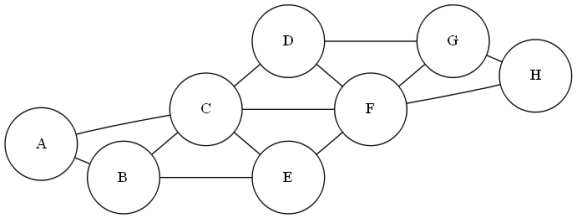
7.8) (2pts)



A. No change from MRV and LCV
B. Change only from MRV
C. Change only from LCV
D. Change from both

MRV is a variable ordering scheme while LCV is a value ordering scheme. As a result, LCV does not change the number of nodes expanded because since our algorithm finds ALL possible assignments, we need to check the value for each variable anyway, so adding LCV will have no benefit or detriment. On the other hand, MRV could change the number of nodes expanded since it changes the order of the search.

7.9) (2pts)



A. No change from MRV and LCV

B. Change only from MRV

C. Change only from LCV

D. Change from both

MRV is a variable ordering scheme while LCV is a value ordering scheme. As a result, LCV does not change the number of nodes expanded because since our algorithm finds ALL possible assignments, we need to check the value for each variable anyway, so adding LCV will have no benefit or detriment. On the other hand, MRV could change the number of nodes expanded since it changes the order of the search.