Lab 3

Tasnimul Hasnat

190041113

Saturday, Sep 30 2023

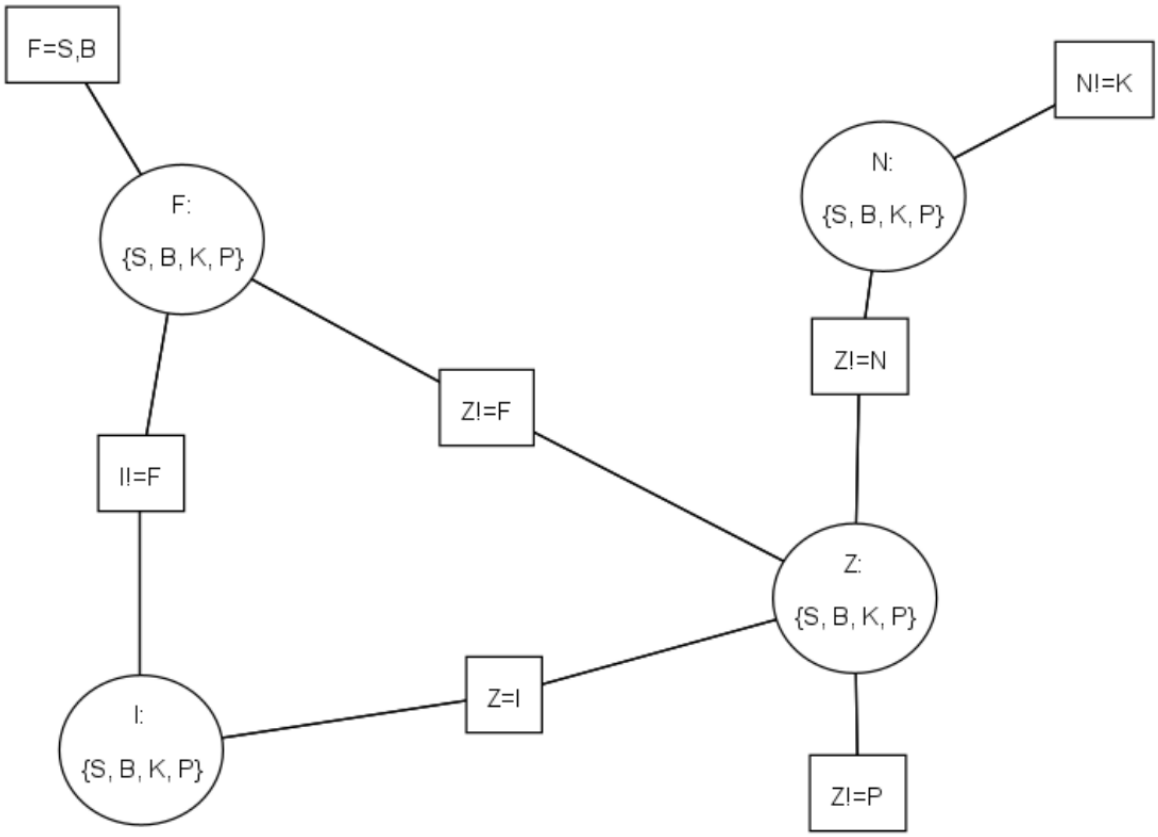
Task 1

The first question had some constraints listed. We were supposed to create a CSP for the problem and explore the solutions.

For this I took 4 variables for each of the persons with domain type string where the domain had initials of all the available food options they had initially. Then I gave constraints for all of them.

The first constraint is that Zahid (Z) does not like paratha. So I gave the unary constraint Z!=P where I unticked Paratha (P) and connected it to Zahid (Z). Then I gave the constraint I!=F where I unticked (S,S), (B,B), (K,K) and (P,P) to ensure Ishrak (I) and Farabi (F) ordered different dishes and connected it to both F and I. Then for Farabi (F) I created a unary constraint with only S,B ticked, named F=S,B and connected it to F because he will order only rice items. Zahid (Z) will copy Ishrak (I) thus giving the constraint Z=I between them with only (S,S), (B,B), (K,K) and (P,P) ticked. But Zahid (Z) will order differently from others. So I gave constraint Z!=F between Farabi (F) and Z and Z!=N between Nafisa (N) and Z. Both with (S,S), (B,B), (K,K) and (P,P) unticked. For Nafisa (N) I gave an unary constraint N!=K as she won't order Kashmiri Naan (K).

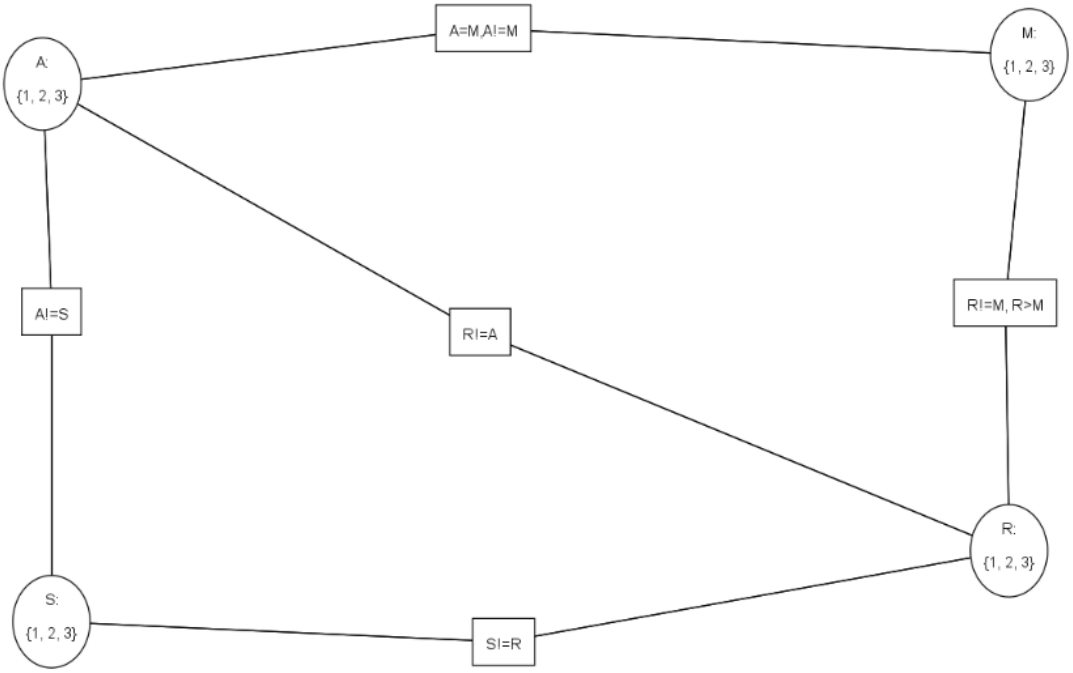
Thus the CSP looked like this:



Task 2

The question had some constraints listed. We were supposed to create a CSP for the problem and explore the solutions.
For this I took 4 variables for each of the persons with domain type integers where the domain consisted of the available floor options they had initially. Then I gave constraints for all of them.
The first constraint I gave is A!=S with (1,1), (2,2), (3,3) unticked and connected to A and S to ensure Ali (A) and Srishty (S) don't live on the same floor. Then I gave a combined constraint for A!=M and A=M with (1,1), (1,2), (2,1), (3,3) unticked to ensure if Ali and Maliha live on the same floor, they live on floor 2 and if Ali and Maliha live on different floors, one of them live on floor 3. Then I gave constraints R!=A and R!=S with (1,1), (2,2), (3,3) unticked and connected between R and A, R and S to ensure Rafid (R) lives on a different floor than Ali A) and Srishty (S). I also gave a combined constraint for R!=M and R>M with only (2,1), (3,1), (3,2) ticked and connected to R and M to ensure Rafid(R) and Maliha(M) live on different floors where Rafid lives on a higher floor than Maliha.

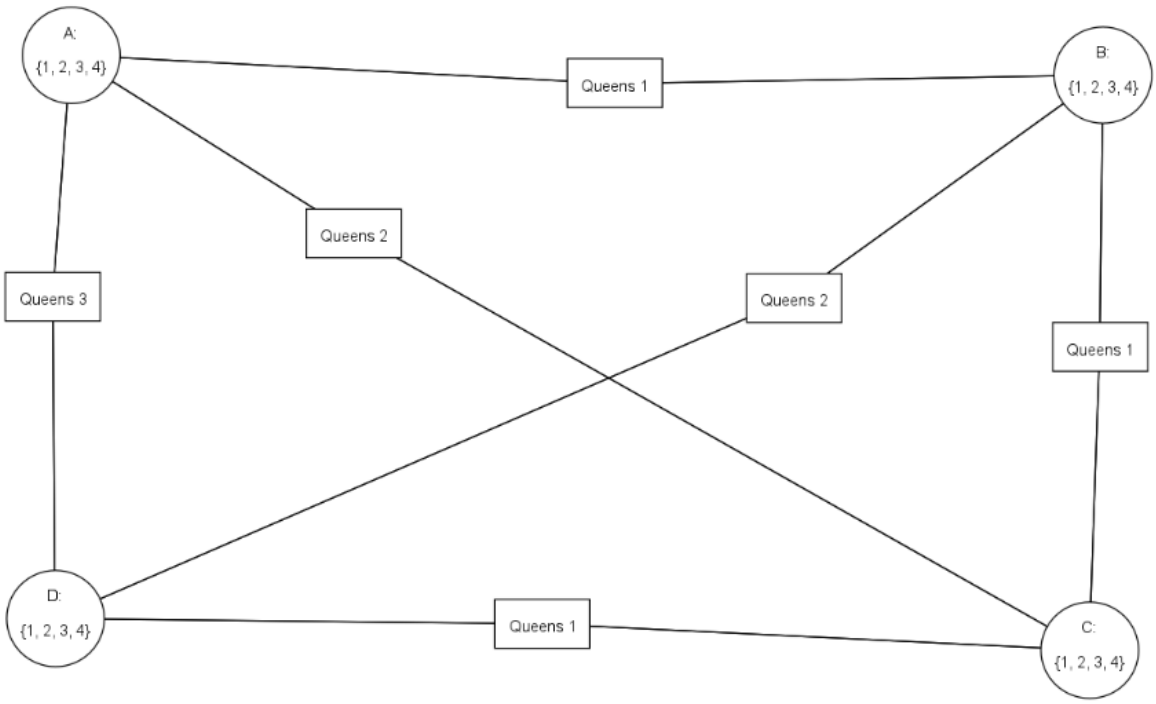
Thus the CSP looked like this:



Task 3

The third problem was a four queens puzzle problem in chess. I took 4 variables named A,B,C,D with domain type integer and domain 1,2,3,4 as it's a 4x4 chess board with 4 chess queens. Then I used the queens type constraint between each of them with the distance 1 between A & B, B & C, C & D, distance 2 between A & C, B & D, distance 3 between A & D.

Thus the CSP looked like this:



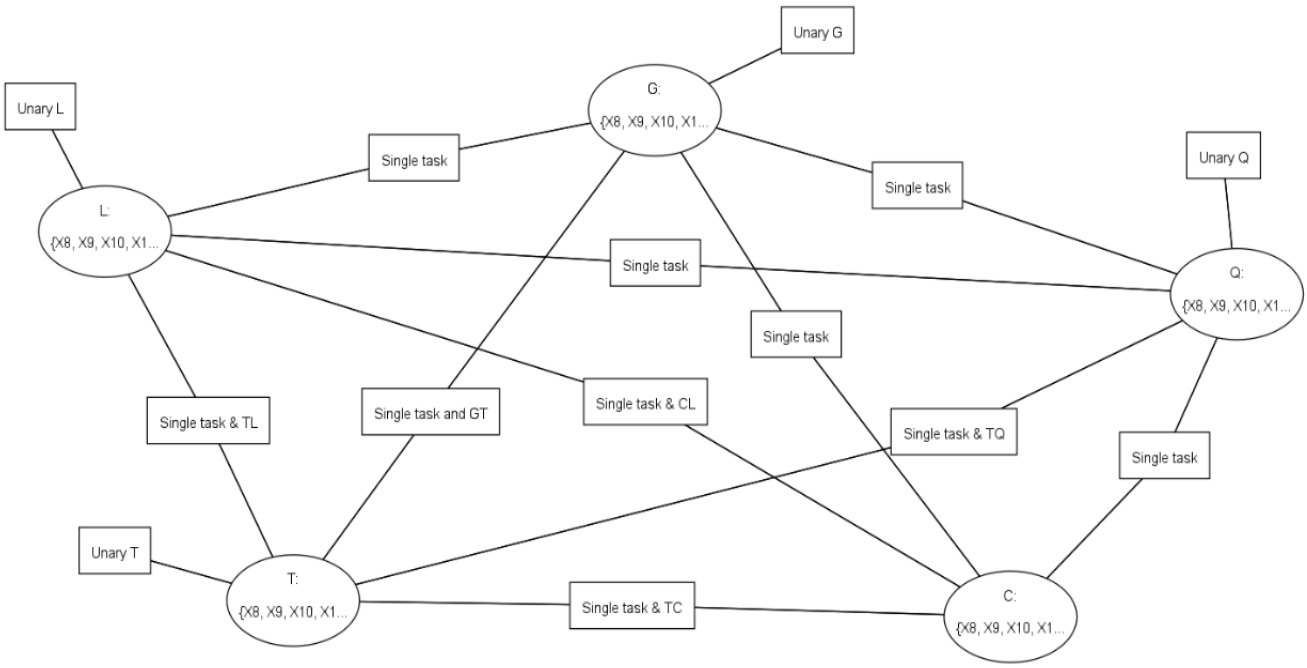
Task 4

The question had some constraints listed. We were supposed to create a CSP for the problem and explore the solutions.
For this I took 5 variables for each of the tasks with domain type string where the domain consisted of the available hours for X & Y initially which are denoted as X8,X9,X10,X11 and Y8,Y9,Y10,Y11. Then I gave constraints for all of them.

For the first constraint to be satisfied, which is each faculty member can do at most 1 task at a time, I gave constraint named Single task between each pair of the variables where I unticked (X8,X8), (X9,X9), (X10,X10), (X11,X11), (Y8,Y8), (Y9,Y9), (Y10,Y10), (Y11,Y11) to ensure X and Y cant do more than 1 task at a time. Thus I have (5*4)/2=10 constraints now. The 2nd Constraint is that C must happen before L. For that I modified the Single task constraint between C & L and named it Single task & CL where I kept the combinations ticked for L & C which satisfies the constraint such as (X9,X8), (X10,X8), (X10,X9) etc. Similarly for the 3rd constraint which requires G must happen T, I modified the Single task constraint between G & T and named it Single task & GT where I kept the combinations ticked for G & T which satisfies the constraint such as (X9,X8) , (X9,Y8) etc. The 4th

constraint is an unary constraint which requires T must be done before 10. For this I gave the constraint Unary T with X10,X11,Y10,Y11
unticked and connected it to T. For the 5th constraint I created an unary constraint and connected to G, named Unary G and kept X8, X9, X10, X11 ticked only as X will do this task. The next constraint is that other faculty members not conducting T must attend the lab and can't do anything else. For this I modified every constraint connection of T with other variables where I unticked every combination that denotes when T is done, any other task is also done such as
(X8,Y8), (X9,Y9), (X10,Y10) etc. In this process I named Single task constraint between T & Q, T & C to Single task & TQ & Single task & TC. The person doing T doesn't do L. For this I modified the Single task constraint between T & L and named it Single task & TL where I unticked the combinations for T & L where both of them are X or L. The person doing L must do
C. For this in the constraint Single task & CL, I unticked every combination for C & L where one of them is X and the other one is Y. Then Q must start at or before 10. For this I created the constraint Unary Q with X11 and Y11 unticked and connected it to Q. Again L must start at or before 10. For this I again created the constraint Unary L with X11 and Y11 unticked and connected it to L.

Thus the CSP looked like this:



After auto-solving I got the following results:

G: {X8}

Q: {X10}

C: {Y8}

T: {X9}

L: {Y10}

This solution meets every constraint mentioned in the question.

—xxx—