

Assignment #2

Brandon
Angeles
Zaid
Habibi

Q.1) Constraint satisfaction

- a)
1. $A=1, D=1, E=1, B=1$
 2. $A=1, D=1, E=1, B=2, C=1, F=1$ Failure
 3. $A=1, D=1, E=1, B=2, C=1, F=2$ Failure
 4. $A=1, D=1, E=1, B=2, C=1, F=3$ Failure
 5. $A=1, D=1, E=1, B=2, C=1, F=4$ Failure
 6. $A=1, D=1, E=1, B=2, C=1, F=5$ Failure
 7. $A=1, D=1, E=1, B=2, C=1, F=6$ Failure
 8. $A=1, D=1, E=1, B=2, C=2$ Failure
 9. $A=1, D=1, E=1, B=2, C=3$ Failure
 10. $A=1, D=1, E=1, B=2, C=4$ Failure
 11. $A=1, D=1, E=1, B=2, C=5$ Failure
 12. $A=1, D=1, E=1, B=2, C=6$ Failure
 13. $A=1, D=1, E=1, B=3, C=1, F=1$ Failure
 14. $A=1, D=1, E=1, B=3, C=1, F=2$ Failure
 15. $A=1, D=1, E=1, B=3, C=1, F=3$ Failure
 16. $A=1, D=1, E=1, B=3, C=1, F=4$ Failure
 17. $A=1, D=1, E=1, B=3, C=1, F=5$ Failure
 18. $A=1, D=1, E=1, B=3, C=1, F=6$ Failure
 19. $A=1, D=1, E=1, B=3, C=2$ Failure
 20. $A=1, D=1, E=1, B=3, C=3$ Failure

- b)
1. $A=1$ Failure
 2. $A=2, F=1$ Failure
 3. $A=3, C=3, F=2, D=2, E=1, B=4$ solution

$$\begin{aligned}
 C) \quad A &= 2, 3, 4, 5 \\
 B &= 2, 3, 4, 5, 6 \\
 C &= 2, 3, 4, 5 \\
 D &= 2, 3, 4, 5 \\
 E &= 1, 2, 3, 4, 5 \\
 F &= 2, 3, 4, 5
 \end{aligned}$$

0.2) Problem Encoding and Propositional Logic

a) First clause we need are the clauses to check each pigeon is put into a hole.

$$T = \bigwedge \{ x_{i,1} \vee x_{i,2} \vee x_{i,3} \vee \dots \vee x_{i,n} \mid 1 \leq i \leq n+1 \}$$

a total of $n+1$ clauses need to be checked

Our next clauses are to check that there is only one pigeon to a hole. In this case we check there is either no pigeon or one pigeon to a hole.

$$F_{1 \leq j \leq n} = \bigwedge \{ \neg x_{i,j} \vee \neg x_{k,j} \mid 1 \leq i \leq k \leq n+1 \}$$

we must check n^2 clauses for the holes

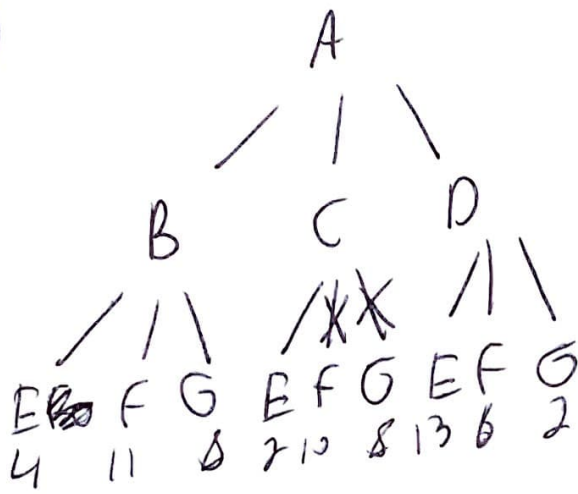
$$\phi = T \wedge F_1 \wedge F_2 \wedge \dots \wedge F_n$$

b) The total number of clauses would be $O(n^3)$

0.3) Adversarial Search

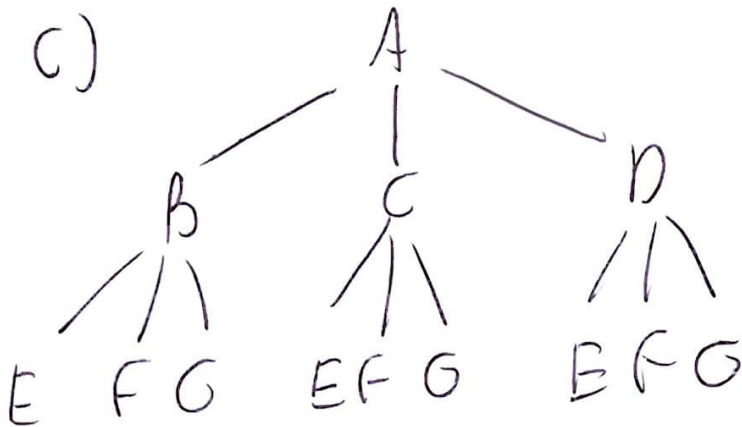
a) The next move for the maximizer should be moving to the left hole, B. $A \rightarrow B$

b)



The node that can be pruned
is from node C the nodes F and G,
from left-to-right pruning

c)



No node can be pruned
from right-to-left pruning