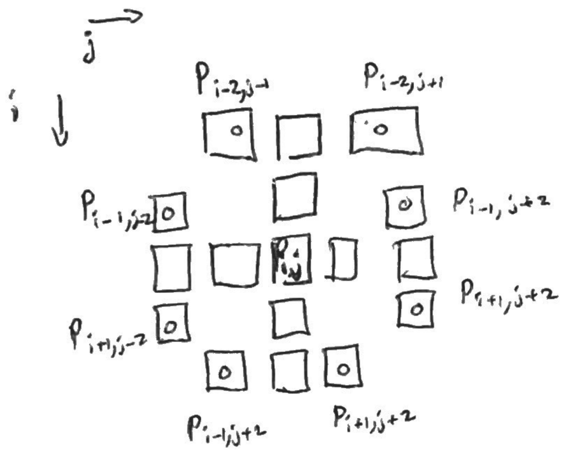
Question 4 Introduction to Artificial Intelligence [redacted]

(a) The variables are boolean, stating whether a pawn/knight is placed on the board or not. There is a variable for every grid on the board. The possible values are P or $\neg P$ (1 or 0) where P (or 1) is a pawn that is placed, and $\neg P$ (or 0) is a pawn that cannot be placed on the board, so it must be left empty. The location of a variable is given by the subscript under the variable P : $P_{i,j}$ for a matrix (ith row, jth column).

(b)



Constraints:

$$P_{i,j} \rightarrow \neg P_{i-2,j-1} \wedge \neg P_{i-2,j+1} \wedge \neg P_{i-1,j-2} \wedge \neg P_{i+1,j-2} \wedge \neg P_{i-1,j+2} \wedge \neg P_{i+1,j+2} \wedge \neg P_{i-1,j+2} \wedge \neg P_{i+1,j+2}$$

$$\neg P_{i,j} \rightarrow P_{i-2,j-1} \vee P_{i-2,j+1} \vee P_{i-1,j-2} \vee P_{i+1,j-2} \vee P_{i-1,j+2} \vee P_{i+1,j+2} \vee P_{i+1,j+2} \vee P_{i-1,j+2}$$

(c) The CSF algorithm (Backtracking Algorithm) can be used to place the pawns in such a way that there would be no conflict, given the constraints, with the arrangement.
Much like the n-queens problem, to solve this problem it's best to place many random pieces on the board until all constraints are in conflict, then backtrack or simply backtrack after initially placing all the pawns on the board. When deciding where to place a pawn, we may use the Min-Conflict algorithm variation (page 225) to calculate the heuristic of a pawn placement based on the number of conflicts that arise as a result for any single placement. For example, a pawn placement where a conflict of 0 versus a conflict of 3 (where 3 constraints are violated) arises, the clear choice would be the placement on the grid with a heuristic of zero.