Q-1) **<u>Sammy's Sport Shop Problem:</u>**

a)- Propositional Knowledge Base:

1. Inferences from the given information
   a. $O1Y \rightarrow C1Y \lor C1B$ (c),
   b. $O1W \rightarrow C1W \lor C1B$
   c. $O2Y \rightarrow C2Y \lor C2B$,
   d. $O2W \rightarrow C2W \lor C2B$ (d)
   e. $O3Y \rightarrow C3Y \lor C3B$,
   f. $O3W \rightarrow C3W \lor C3B$ (f)

2. Labels displayed wrong
   a. $L1Y \rightarrow \neg C1Y$,
   b. $L1W \rightarrow \neg C1W$,
   c. $L1B \rightarrow \neg C1B$
   d. $L2Y \rightarrow \neg C2Y$,
   e. $L2W \rightarrow \neg C2W$,
   f. $L2B \rightarrow \neg C2B$
   g. $L3Y \rightarrow \neg C3Y$,
   h. $L3W \rightarrow \neg C3W$,
   i. $L3B \rightarrow \neg C3B$ (a)

3. No Two boxes should have the same contents
   a. $C1Y \rightarrow \neg C2Y \land \neg C3Y$,
   b. $C1W \rightarrow \neg C2W \land \neg C3W$,
   c. $C1B \rightarrow \neg C2B \land \neg C3B$ (e)
   d. $C2Y \rightarrow \neg C1Y \land \neg C3Y$,
   e. $C2W \rightarrow \neg C1W \land \neg C3W$,
   f. $C2B \rightarrow \neg C1B \land \neg C3B$
   g. $C3Y \rightarrow \neg C2Y \land \neg C1Y$ (b),
   h. $C3W \rightarrow \neg C2W \land \neg C1W$,
   i. $C3B \rightarrow \neg C2B \land \neg C1B$

4. There should be at least One box of each color
   a. $C1Y \lor C1W \lor C1B$,
   b. $C2Y \lor C2W \lor C2B$,
   c. $C3Y \lor C3W \lor C3B$

b)- Natural Deduction to prove (KB |= C2W) that box 2 contains white tennis balls

5. From O3Y and (1.e), using Modus Ponens we get C3Y ∨ C3B
6. From L3B and (2.i), using Modus Ponens we get ¬C3B
7. From 5 and 6, using resolution we get C3Y.
8. From 7 and (3.g), using Modus Ponens we get ¬C1Y ∧ ¬C2Y
9. From O1Y and (1.a), using Modus Ponens we get C1Y v C1B
10. From 8, using AND Elimination we get ¬C1Y
11. From 9 and 10, using resolution we get C1B
12. From O2W and (1.d), using Modus Ponens we get C2W ∨ C2B
13. From 11 and (3.e), using Modus Ponens we get ¬C2B ∧ ¬C3B
14. From 13, using AND Elimination ¬C2B
15. From 12 and 14, using resolution we get C2W

Hence **KB |= C2W**

c)- Show that box 2 must contain white balls via a Resolution Refutation proof

// just the sentences I need...

16. From (1.e), using Implication Elimination we get ¬O3Y ∨ C3Y ∨ C3B (c0)
17. From (1.a), using Implication Elimination we get ¬O1Y ∨ C1Y ∨ C1B (c1)
18. From (1.d), using Implication Elimination we get ¬O2W ∨ C2W ∨ C2B (c2)
19. From (2.i), using Implication Elimination we get ¬L3B ∨ ¬C3B (c3)
20. From (3.c), using Implication Elimination ¬C1B ∨ ¬C2B (c4a) , ¬C1B ∨ ¬C3B (c4b)
21. From (3.g), using Implication Elimination ¬C3Y ∨ ¬C2Y (c5a) , ¬C3Y ∨ ¬C1Y (c5b)

The Facts become Unit Clauses as below

22. O1Y
23. O2W
24. O3Y
25. L1W
26. L2Y
27. L3B
28. **¬C2W // considering negation of query as per refutation (Contradiction)**
29. Using resolution on c0, 24 we get C3Y ∨ C3B
30. Using resolution on c3, 27 we get ¬C3B
31. Using resolution on 29, 30 we get C3Y
32. Using resolution on c5b, 31 we get ¬C1Y
33. Using resolution on c1, 22 we get C1Y ∨ C1B
34. Using resolution on 32, 33 we get C1B
35. Using resolution on c4a, 34 we get ¬C2B
36. Using resolution on c2, 23 we get C2W ∨ C2B
37. Using resolution on 35, 36 we get C2W
38. Using resolution on 28, 37 we get an empty clause.

So thus **KB |= C2W. Hence Box 2 must contain White Balls.**

Q-2) <u>4-Queens problem:</u>

For this problem, the relevant propositional symbols are:

$Q_{A,1}$ $Q_{A,2}$ $Q_{A,3}$ $Q_{A,4}$ $Q_{B,1}$ $Q_{B,2}$ $Q_{B,3}$ $Q_{B,4}$ $Q_{C,1}$ $Q_{C,2}$ $Q_{C,3}$ $Q_{C,4}$ $Q_{D,1}$ $Q_{D,2}$ $Q_{D,3}$ $Q_{D,4}$

With these 16 symbols, there are 2^{16} possible models.

Conditions:

1. There is atleast one queen in each column
   a. $Q_{A,1} \vee Q_{A,2} \vee Q_{A,3} \vee Q_{A,4}$
   b. $Q_{B,1} \vee Q_{B,2} \vee Q_{B,3} \vee Q_{B,4}$
   c. $Q_{C,1} \vee Q_{C,2} \vee Q_{C,3} \vee Q_{C,4}$
   d. $Q_{D,1} \vee Q_{D,2} \vee Q_{D,3} \vee Q_{D,4}$

2. No two Queens in the same row
   a. $\neg Q_{A,1} \vee \neg Q_{B,1}$
   b. $\neg Q_{A,1} \vee \neg Q_{C,1}$
   c. $\neg Q_{A,1} \vee \neg Q_{D,1}$
   d. $\neg Q_{B,1} \vee \neg Q_{C,1}$
   e. $\neg Q_{B,1} \vee \neg Q_{D,1}$
   f. $\neg Q_{C,1} \vee \neg Q_{D,1}$
   g. $\neg Q_{A,2} \vee \neg Q_{B,2}$
   h. $\neg Q_{A,2} \vee \neg Q_{C,2}$
   i. $\neg Q_{A,2} \vee \neg Q_{D,2}$
   j. $\neg Q_{B,2} \vee \neg Q_{C,2}$
   k. $\neg Q_{B,2} \vee \neg Q_{D,2}$
   l. $\neg Q_{C,2} \vee \neg Q_{D,2}$
   m. $\neg Q_{A,3} \vee \neg Q_{B,3}$
   n. $\neg Q_{A,3} \vee \neg Q_{C,3}$
   o. $\neg Q_{A,3} \vee \neg Q_{D,3}$
   p. $\neg Q_{B,3} \vee \neg Q_{C,3}$
   q. $\neg Q_{B,3} \vee \neg Q_{D,3}$
   r. $\neg Q_{C,3} \vee \neg Q_{D,3}$
   s. $\neg Q_{A,4} \vee \neg Q_{B,4}$
   t. $\neg Q_{A,4} \vee \neg Q_{C,4}$
   u. $\neg Q_{A,4} \vee \neg Q_{D,4}$
   v. $\neg Q_{B,4} \vee \neg Q_{C,4}$
   w. $\neg Q_{B,4} \vee \neg Q_{D,4}$
   x. $\neg Q_{C,4} \vee \neg Q_{D,4}$

3. No two Queens in the same column
   a. $\neg Q_{A,1} \vee \neg Q_{A,2}$
   b. $\neg Q_{A,1} \vee \neg Q_{A,3}$
   c. $\neg Q_{A,1} \vee \neg Q_{A,4}$
   d. $\neg Q_{A,2} \vee \neg Q_{A,3}$
   e. $\neg Q_{A,2} \vee \neg Q_{A,4}$
   f. $\neg Q_{A,3} \vee \neg Q_{A,4}$
   g. $\neg Q_{B,1} \vee \neg Q_{B,2}$
   h. $\neg Q_{B,1} \vee \neg Q_{B,3}$
   i. $\neg Q_{B,1} \vee \neg Q_{B,4}$
   j. $\neg Q_{B,2} \vee \neg Q_{B,3}$
   k. $\neg Q_{B,2} \vee \neg Q_{B,4}$
   l. $\neg Q_{B,3} \vee \neg Q_{B,4}$
   m. $\neg Q_{C,1} \vee \neg Q_{C,2}$
   n. $\neg Q_{C,1} \vee \neg Q_{C,3}$
   o. $\neg Q_{C,1} \vee \neg Q_{C,4}$
   p. $\neg Q_{C,2} \vee \neg Q_{C,3}$
   q. $\neg Q_{C,2} \vee \neg Q_{C,4}$
   r. $\neg Q_{C,3} \vee \neg Q_{C,4}$
   s. $\neg Q_{D,1} \vee \neg Q_{D,2}$
   t. $\neg Q_{D,1} \vee \neg Q_{D,3}$
   u. $\neg Q_{D,1} \vee \neg Q_{D,4}$
   v. $\neg Q_{D,2} \vee \neg Q_{D,3}$
   w. $\neg Q_{D,2} \vee \neg Q_{D,4}$
   x. $\neg Q_{D,3} \vee \neg Q_{D,4}$

4. No more than one queen in each diagonal

   a. $\neg Q_{A,1} \vee \neg Q_{B,2}$
   b. $\neg Q_{A,1} \vee \neg Q_{C,3}$
   c. $\neg Q_{A,1} \vee \neg Q_{D,4}$
   d. $\neg Q_{A,2} \vee \neg Q_{B,1}$
   e. $\neg Q_{A,2} \vee \neg Q_{B,3}$
   f. $\neg Q_{A,2} \vee \neg Q_{C,4}$
   g. $\neg Q_{C,2} \vee \neg Q_{D,3}$
   h. $\neg Q_{A,3} \vee \neg Q_{B,2}$
   i. $\neg Q_{A,3} \vee \neg Q_{C,1}$
   j. $\neg Q_{A,3} \vee \neg Q_{B,4}$
   k. $\neg Q_{A,4} \vee \neg Q_{B,3}$
   l. $\neg Q_{A,4} \vee \neg Q_{C,2}$
   m. $\neg Q_{A,4} \vee \neg Q_{D,1}$
   n. $\neg Q_{C,3} \vee \neg Q_{D,2}$
   o. $\neg Q_{B,1} \vee \neg Q_{C,2}$
   p. $\neg Q_{B,1} \vee \neg Q_{D,3}$
   q. $\neg Q_{B,2} \vee \neg Q_{C,1}$
   r. $\neg Q_{B,2} \vee \neg Q_{C,3}$
   s. $\neg Q_{B,2} \vee \neg Q_{D,4}$
   t. $\neg Q_{B,3} \vee \neg Q_{C,2}$
   u. $\neg Q_{C,3} \vee \neg Q_{D,4}$
   v. $\neg Q_{B,3} \vee \neg Q_{D,1}$
   w. $\neg Q_{B,3} \vee \neg Q_{C,4}$
   x. $\neg Q_{B,4} \vee \neg Q_{C,3}$
   y. $\neg Q_{B,4} \vee \neg Q_{D,2}$
   z. $\neg Q_{C,1} \vee \neg Q_{D,2}$
   z1. $\neg Q_{C,2} \vee \neg Q_{D,1}$
   z2. $\neg Q_{C,4} \vee \neg Q_{D,3}$

The total number of clauses are 4 + 24 + 24 + 28 = 80

The 4 queens problem has a solution if and only if this 80 clause formula is Satisfiable. If an algorithm not only decides that the formula is satisfiable, but also returns a model, the truth values in the model tells us where to place the queens.

DPLL algorithm is used for finding a satisfying model for this puzzle.

The Symbols in the next page have the following meaning:

ns – Information Not sufficient, call DPLL again to bind next variable

r - No two queens in the same row (rules in 2 belongs to this class)

d - No two queens on the same diagonal (rules in 4 belongs to this class)

c - Exactly One queen in one column (rules in 3 belongs to this class)

red - start point of back tracking

green - ending point of backtracking

Last column in the table indicates the rules that is violated.

For example, (1.a) means rule 1a is violated because of that assignment.

a)- Next page shows, DPLL Algorithm without Heuristics.

$Q_{A,1}$	$Q_{A,2}$	$Q_{A,3}$	$Q_{A,4}$	$Q_{B,1}$	$Q_{B,2}$	$Q_{B,3}$	$Q_{B,4}$	$Q_{C,1}$	$Q_{C,2}$	$Q_{C,3}$	$Q_{C,4}$	$Q_{D,1}$	$Q_{D,2}$	$Q_{D,3}$	$Q_{D,4}$	#Failed
T																ns
T	T															c (3.a)
T	F															ns
T	F	T														c (3.a)
T	F	F														ns
T	F	F	T													c (3.a)
T	F	F	F													ns
T	F	F	F	T												r (2.a)
T	F	F	F	F												ns
T	F	F	F	F	T											d (4.a)
T	F	F	F	F	F											ns
T	F	F	F	F	F	T										ns
T	F	F	F	F	F	T	T									c (3.i)
T	F	F	F	F	F	T	F									ns
T	F	F	F	F	F	T	F	T								r (2.b)
T	F	F	F	F	F	T	F	F								ns
T	F	F	F	F	F	T	F	F	T							d (4.t)
T	F	F	F	F	F	T	F	F	F							ns
T	F	F	F	F	F	T	F	F	F	T						r (2.p)
T	F	F	F	F	F	T	F	F	F	F						ns
T	F	F	F	F	F	T	F	F	F	F	T					d (4.w)
T	F	F	F	F	F	T	F	F	F	F	F					c (1.c)
T	F	F	F	F	F	F										ns
T	F	F	F	F	F	F	T									ns
T	F	F	F	F	F	F	T	T								r (2.b)
T	F	F	F	F	F	F	T	F								ns
T	F	F	F	F	F	F	T	F	T							ns
T	F	F	F	F	F	F	T	F	T	T						c (3.p)
T	F	F	F	F	F	F	T	F	T	F						ns
T	F	F	F	F	F	F	T	F	T	F	T					c (3.q)
T	F	F	F	F	F	F	T	F	T	F	F					ns
T	F	F	F	F	F	F	T	F	T	F	F	T				r (2.c)
T	F	F	F	F	F	F	T	F	T	F	F	F				ns
T	F	F	F	F	F	F	T	F	T	F	F	F	T			r (2.l)
T	F	F	F	F	F	F	T	F	T	F	F	F	F			ns
T	F	F	F	F	F	F	T	F	T	F	F	F	F	T		d (4.g)
T	F	F	F	F	F	F	T	F	T	F	F	F	F	F		ns
T	F	F	F	F	F	F	T	F	T	F	F	F	F	F	T	d (4.c)
T	F	F	F	F	F	F	T	F	T	F	F	F	F	F	F	c (1.d)
F																ns
F	T															ns
F	T	T														c (3.d)
F	T	F														ns
F	T	F	T													c (3.e)
F	T	F	F													ns
F	T	F	F	T												d (4.d)
F	T	F	F	F												ns

$Q_{A,1}$	**$Q_{A,2}$**	**$Q_{A,3}$**	**$Q_{A,4}$**	**$Q_{B,1}$**	**$Q_{B,2}$**	**$Q_{B,3}$**	**$Q_{B,4}$**	**$Q_{C,1}$**	**$Q_{C,2}$**	**$Q_{C,3}$**	**$Q_{C,4}$**	**$Q_{D,1}$**	**$Q_{D,2}$**	**$Q_{D,3}$**	**$Q_{D,4}$**	**#Failed**
F	T	F	F	F	T											r (2.g)
F	T	F	F	F	F											ns
F	T	F	F	F	F	T										d (4.e)
F	T	F	F	F	F	F										ns
F	T	F	F	F	F	F	T									ns
F	T	F	F	F	F	F	T	T								ns
F	T	F	F	F	F	F	T	T	T							c (3.m)
F	T	F	F	F	F	F	T	T	F							ns
F	T	F	F	F	F	F	T	T	F	T						c (3.n)
F	T	F	F	F	F	F	T	T	F	F						ns
F	T	F	F	F	F	F	T	T	F	F	T					c (3.o)
F	T	F	F	F	F	F	T	T	F	F	F					ns
F	T	F	F	F	F	F	T	T	F	F	F	T				r (2.f)
F	T	F	F	F	F	F	T	T	F	F	F	F				ns
F	T	F	F	F	F	F	T	T	F	F	F	F	T			r (2.i)
F	T	F	F	F	F	F	T	T	F	F	F	F	F			ns
F	T	F	F	F	F	F	T	T	F	F	F	F	F	T		ns
F	T	F	F	F	F	F	T	T	F	F	F	F	F	T	T	c (3.x)
F	**T**	**F**	**F**	**F**	**F**	**F**	**T**	**T**	**F**	**F**	**F**	**F**	**F**	**T**	**F**	Success

Hence the below model is derived using DPLL algorithm.

$Q_{A,1}$	**$Q_{A,2}$**	**$Q_{A,3}$**	**$Q_{A,4}$**	**$Q_{B,1}$**	**$Q_{B,2}$**	**$Q_{B,3}$**	**$Q_{B,4}$**	**$Q_{C,1}$**	**$Q_{C,2}$**	**$Q_{C,3}$**	**$Q_{C,4}$**	**$Q_{D,1}$**	**$Q_{D,2}$**	**$Q_{D,3}$**	**$Q_{D,4}$**
F	T	F	F	F	F	F	T	T	F	F	F	F	F	T	F

It can be shown as below:

		$Q_{C,1}$	
$Q_{A,2}$			
			$Q_{C,3}$
	$Q_{B,4}$		

b)- DPLL with PureSymbol and UnitClause Heuristic is as below:

$Q_{A,1}$	$Q_{A,2}$	$Q_{A,3}$	$Q_{A,4}$	$Q_{B,1}$	$Q_{B,2}$	$Q_{B,3}$	$Q_{B,4}$	$Q_{C,1}$	$Q_{C,2}$	$Q_{C,3}$	$Q_{C,4}$	$Q_{D,1}$	$Q_{D,2}$	$Q_{D,3}$	$Q_{D,4}$	#Heur
T																UCH
T	F	F	F	F	F			F		F		F			F	ns
T	F	F	F	F	F	T		F		F		F			F	UCH
T	F	F	F	F	F	T	F	F	F	F	F	F		F	F	1 C
T	F	F	F	F	F	F		F		F		F			F	UCH 1a
T	F	F	F	F	F	F	T	F		F		F			F	UCH
T	F	F	F	F	F	F	T	F		F	F	F	F		F	UCH 1c,1d
T	F	F	F	F	F	F	T	F	T	F	F	F	F	T	F	4.g
F																Ns
F	T															UCH
F	T	F	F	F	F	F			F		F		F			UCH 1b
F	T	F	F	F	F	F	T		F		F		F			UCH
F	T	F	F	F	F	F	T		F	F	F		F		F	UCH 1c
F	T	F	F	F	F	F	T	T	F	F	F		F		F	UCH
F	T	F	F	F	F	F	T	T	F	F	F	F	F		F	PSH
F	**T**	**F**	**F**	**F**	**F**	**F**	**T**	**T**	**F**	**F**	**F**	**F**	**F**	**T**	**F**	Success

UCH – Unit Clause Heuristic will be used at that step

PSH – Pure Symbol Heuristic will be used at that step

ns- Not sufficient assignments so call DPLL again to bind other literal

Steps Involved:

1. No pure symbol/unit clause initially so randomly assign $Q_{A,1}$ as True
2. Above assignment will create multiple Unit Clauses- 2a, 2b, 2c, 3a, 3b, 3c, 4a, 4b, 4c
3. These assignments are not sufficient (ns) so call DPLL again to bind other literal $Q_{B,3}$ as True
4. Above assignment will create additional multiple Unit Clauses – 2q, 2l, 2w, 2t
5. Above assignment will **violate** the constraint 1c. So we need to **Backtrack**. Set $Q_{B,3}$ as False.
6. Above assignment will create additional Unit Clause- 1b
7. Above assignment will create additional Unit Clauses – 2v, 4y
8. Above assignment will create additional Unit Clauses- 1c, 1d
9. Above assignments will **violate** the constraint 4g. So we need to **Backtrack**. Set $Q_{A,1}$ as False.
10. This assignment is not sufficient (ns) so call DPLL again to bind other literal $Q_{A,2}$ as True
11. Above assignment will create additional multiple Unit Clauses – 2g, 2h, 2i, 3d, 3e, 4d, 4e, 4f
12. Above assignment will create additional multiple Unit Clause – 1b
13. Above assignment will create additional Unit Clauses – 4x, 4y
14. Above assignment will create additional Unit Clause- 1c
15. Above assignment will create additional Unit Clause- 2f
16. Above assignment will create additional Pure Symbol(It's also a Unit Class) – 1d
17. Resulting Model will satisfy all the constraints. **Hence it's our required Model.**

Q-3) - Tic- Tac- Toe Game Puzzle:

For this problem, the relevant propositional symbols are:

X11 X12 X13 X21 X22 X23 X31 X32 X33 O11 O12 O13 O21 O22 O23 O31 O32 O33

?11 ?12 ?13 ?21 ?22 ?23 ?31 ?32 ?33

moveX11 moveX12 moveX13 moveX21 moveX22 moveX23 moveX31 moveX32 moveX33

moveO11 moveO12 moveO13 moveO21 moveO22 moveO23 moveO31 moveO32 moveO33

canWinX11 canWinX12 canWinX13 canWinX21 canWinX22 canWinX23 canWinX31 canWinX32 canWinX33

canWinO11 canWinO12 canWinO13 canWinO21 canWinO22 canWinO23 canWinO31 canWinO32 canWinO33

forcedMoveX11 forcedMoveX12 forcedMoveX13 forcedMoveX21 forcedMoveX22 forcedMoveX23 forcedMoveX31 forcedMoveX32 forcedMoveX33

canWinX

KnowledgeBase:

1. **When X can win:**
   a. X11 ∧ X12 ∧ ?13 →canWinX13
   b. X11 ∧ X22 ∧ ?33 →canWinX33
   c. X11 ∧ X33 ∧ ?22 →canWinX22
   d. X11 ∧ X21 ∧ ?31 →canWinX31
   e. X12 ∧ X13 ∧ ?11 →canWinX11
   f. X12 ∧ X22 ∧ ?32 →canWinX32
   g. X12 ∧ ?22 ∧ X32 →canWinX22
   h. X13 ∧ X23 ∧ ?33 →canWinX33
   i. X13 ∧ X33 ∧ ?23 →canWinX23
   j. X13 ∧ X22 ∧ ?31 →canWinX31
   k. X13 ∧ X31 ∧ ?22 →canWinX22
   l. X21 ∧ X22 ∧ ?23→canWinX23
   m. X21 ∧ X23 ∧ ?22→canWinX22
   n. X22 ∧ X23 ∧ ?21→canWinX21
   o. X22 ∧ X23 ∧ ?21→canWinX21
   p. X31 ∧ X32 ∧ ?33→canWinX33
   q. X31 ∧ X33 ∧ ?32→canWinX32
   r. X31 ∧ X21 ∧ ?11→canWinX11
   s. X31 ∧ X11 ∧ ?21→canWinX21
   t. X31 ∧ X22 ∧ ?13→canWinX13
   u. X32 ∧ X33 ∧ ?31→canWinX31
   v. X32 ∧ X22 ∧ ?12→canWInX12
   w. X31 ∧ X23 ∧ ?13 →canWinX13
   x. X31 ∧ X22 ∧ ?11 →canWinX11

2. **When O can win:**
   a. O11 ∧ O12 ∧ ?13 →canWinO13
   b. O11 ∧ O22 ∧ ?33 →canWinO33
   c. O11 ∧ O33 ∧ ?22 →canWinO22
   d. O11 ∧ O21 ∧ ?31 →canWinO31
   e. O12 ∧ O13 ∧ ?11 →canWinO11
   f. O12 ∧ O22 ∧ ?32 →canWinO32
   g. O12 ∧ ?22 ∧ O32 →canWinO22
   h. O13 ∧ O23 ∧ ?33 →canWinO33
   i. O13 ∧ O33 ∧ ?23 →canWinO23
   j. O13 ∧ O22 ∧ ?31 →canWinO31
   k. O13 ∧ O31 ∧ ?22 →canWinO22
   l. O21 ∧ O22 ∧ ?23→canWinO23
   m. O21 ∧ O23 ∧ ?22→canWinO22
   n. O22 ∧ O23 ∧ ?21→canWinO21
   o. O22 ∧ O23 ∧ ?21→canWinO21
   p. O31 ∧ O32 ∧ ?33→canWinO33
   q. O31 ∧ O33 ∧ ?32→canWinO32
   r. O31 ∧ O21 ∧ ?11→canWinO11
   s. O31 ∧ O11 ∧ ?21→canWinO21
   t. O31 ∧ O22 ∧ ?13→canWinO13
   u. O32 ∧ O33 ∧ ?31→canWinO31
   v. O32 ∧ O22 ∧ ?12→canWinO12
   w. O31 ∧ O23 ∧ ?13→canWinO13
   x. O31 ∧ O22 ∧ ?11→canWinO11

3. **Block a potential win by O here:**
    a. canWinO11 ↔ forcedMoveX11
    b. canWinO12 ↔ forcedMoveX12
    c. canWinO13 ↔ forcedMoveX13
    d. canWinO21 ↔ forcedMoveX21
    e. canWinO22 ↔forcedMoveX22
    f. canWinO23 ↔ forcedMoveX23
    g. canWinO31 ↔ forcedMoveX31
    h. canWinO32 ↔ forcedMoveX32
    i. canWinO33 ↔ forcedMoveX33

4. **Decide optimal move and move X:**
    a. canWinX11 ∨ canWinX12 ∨ canWinX13 ∨ canWinX21 ∨ canWinX22 ∨ canWinX23 ∨ canWinX31 ∨ canWinX32 ∨ canWinX23 → canWinX
    b. canWinX11 → moveX11
    c. canWinX12 → moveX12
    d. canWinX13 → moveX13
    e. canWinX21 → moveX21
    f. canWinX22 → moveX22
    g. canWinX23 → moveX23
    h. canWinX31 → moveX31
    i. canWinX32 → moveX32
    j. canWinX33 → moveX33
    k. forcedMoveX11 ∧ ¬canWinX → moveX11
    l. forcedMoveX12 ∧ ¬canWinX → moveX12
    m. forcedMoveX13 ∧ ¬canWinX → moveX13
    n. forcedMoveX21 ∧ ¬canWinX → moveX21
    o. forcedMoveX22 ∧ ¬canWinX → moveX22
    p. forcedMoveX23 ∧ ¬canWinX → moveX23
    q. forcedMoveX31 ∧ ¬canWinX → moveX31
    r. forcedMoveX32 ∧ ¬canWinX → moveX32
    s. forcedMoveX33 ∧ ¬canWinX → moveX33

Thus, all these rules will infer all desirable moves for **X** based on the given strategies.

<--**Thank You, Patrick** --->