1. **Proofs**
    a. **Prove that "Implication Introduction" (the opposite of Implication Elimination) is a sound rule of inference (ROI) using a truth table. If you have a Horn clause, with 1 positive literal and n-1 negative literals, like (¬X∨Z∨¬Y), you can transform it into a conjunctive rule by collecting the negative literals as positive antecedents, e.g. X∧Y→Z. It is sufficient to prove this for n-1=2 antecedents. (In fact, this is a truth-preserving operation, hence sound.)**

X	**Y**	**Z**	**¬X**	**¬Y**	**¬X∨¬Y∨ Z**	**X∧Y → Z**
0	0	0	1	1	1	1
0	0	1	1	1	1	1
0	1	0	1	0	1	1
0	1	1	1	0	1	1
1	0	0	0	1	1	1
1	0	1	0	1	1	1
1	1	0	0	0	0	0
1	1	1	0	0	1	1

In all cases where ¬X∨¬Y∨Z is true, X∧Y → Z is also true - therefore the implication introduction is a sound rule-of-inference.

b. Prove that (A∧B→C∧D) |- (A∧B→C) ("conjunctive rule splitting") is a sound rule-of inference using a truth table.

A	B	C	D	C∧D	A∧B	A∧B→C∧D	A∧B→C
0	0	0	0	0	0	1	1
0	0	0	1	0	0	1	1
0	0	1	0	0	0	1	1
0	0	1	1	1	0	1	1
0	1	0	0	0	0	1	1
0	1	0	1	0	0	1	1
0	1	1	0	0	0	1	1
0	1	1	1	1	0	1	1
1	0	0	0	0	0	1	1
1	0	0	1	0	0	1	1
1	0	1	0	0	0	1	1
1	0	1	1	1	0	1	1
1	1	0	0	0	1	0	0
1	1	0	1	0	1	0	0
1	1	1	0	0	1	0	1
1	1	1	1	1	1	1	1

In all cases where A∧B→C∧D is true, A∧B→C is also true - therefore conjunctive rule splitting is a sound rule-of-inference.

c. **Also prove $(A \land B \to C \land D) \models (A \land B \to C)$ using Natural Deduction. (hint: use 1a above)**

1. Assume $A \land B$
2. Use AndElimination to get C from $C \land D$.
3. Use Implication Introduction to get $(A \land B \to C)$
4. Therefore $(A \land B \to C \land D) \models (A \land B \to C)$

d. **Also prove $(A \land B \to C \land D) \models (A \land B \to C)$ using Resolution.**

1. Convert the expressions $(A \land B \to C \land D)$ and $(A \land B \to C)$ into CNF
    a. $(A \land B \to C \land D) \equiv \neg(A \land B) \lor (C \land D) \to \neg A \lor \neg B \lor C \land D$
    b. $(A \land B \to C) \equiv \neg(A \land B) \lor C \to \neg A \lor \neg B \lor C$
2. Apply the resolution rule $(A \lor B, \neg A \lor C) \Rightarrow (B \lor C)$ to resolve the CNF clauses
    a. $\neg A \lor \neg B \lor C \land D$
    b. $\neg A \lor \neg B \lor C$
    c. $\neg A \lor \neg B \lor C \land D \lor C$ (Resolution)
3. $\neg A \lor \neg B \lor C$ (Simplification)
4. We have successfully shown that $(A \land B \to C \land D)$ implies $(A \land B \to C)$

2. **Sammy's Sport Shop:**
   You are the proprietor of Sammy's Sport Shop. You have just received a shipment of three boxes filled with tennis balls. One box contains only yellow tennis balls, one box contains only white tennis balls, and one contains both yellow and white tennis balls. You would like to stock the tennis balls in appropriate places on your shelves. Unfortunately, the boxes have been labeled incorrectly; the manufacturer tells you that you have exactly one box of each, but that each box is definitely labeled wrong. You draw one ball from each box and observe its color. Given the initial (incorrect) labeling of the boxes above, and the three observations, use Propositional Logic to infer the correct contents of the middle box.

   Use propositional symbols in the following form: O1Y means a yellow ball was drawn (observed) from box 1, L1W means box 1 was initially labeled white, C1W means box 1 contains (only) white balls, and C1B means box 1 actually contains both types of tennis balls.

   Note, there is no 'O1B', etc, because you can't directly "observe both". When you draw a tennis ball, it will either be white or yellow.

   The initial facts describing this particular situation are: {O1Y, L1W, O2W, L2Y, O3Y, L3B}

a. **Using these propositional symbols, write a propositional knowledge base (sammy.kb) that captures the knowledge in this domain (i.e. implications of what different observations or labels mean, as well as constraints inherent in this problem, such as that all boxes have different contents). Do it in a complete and general way, writing down all the rules and constraints, not just the ones needed to make the specific inference about the middle box. Do not include derived knowledge that depends on the particular labeling of this instance shown above; stick to what is stated in the problem description above. Your KB should be general enough to reason about any alternative scenario, not just the one given above (e.g. with different observations and labels and box contents)**

1. Observation Implications
    a. $O1Y \rightarrow C1Y \lor C1B$, $O1W \rightarrow C1W \lor C1B$
    b. $O2Y \rightarrow C2Y \lor C2B$, $O2W \rightarrow C2W \lor C2B$
    c. $O3Y \rightarrow C3Y \lor C3B$, $O3W \rightarrow C3W \lor C3B$
2. Incorrect Labels Implications
    a. $L1Y \rightarrow \neg C1Y$, $L1W \rightarrow \neg C1W$, $L1B \rightarrow \neg C1B$
    b. $L2Y \rightarrow \neg C2Y$, $L2W \rightarrow \neg C2W$, $L2B \rightarrow \neg C2B$
    c. $L3Y \rightarrow \neg C3Y$, $L3W \rightarrow \neg C3W$, $L3B \rightarrow \neg C3B$
3. Exactly One Box of Each Color Implications
    a. There exists at least 1 box of each color
        i. $C1Y \lor C1W \lor C1B$, $C2Y \lor C2W \lor C2B$, $C3Y \lor C3W \lor C3B$
    b. There exists no more than 1 box of the same contents
        i. $C1Y \rightarrow \neg C2Y \land \neg C3Y$, $C1W \rightarrow \neg C2W \land \neg C3W$, $C1B \rightarrow \neg C2B \land \neg C3B$
        ii. $C2Y \rightarrow \neg C1Y \land \neg C3Y$, $C2W \rightarrow \neg C1W \land \neg C3W$, $C2B \rightarrow \neg C1B \land \neg C3B$
        iii. $C3Y \rightarrow \neg C2Y \land \neg C1Y$, $C3W \rightarrow \neg C2W \land \neg C1W$, $C3B \rightarrow \neg C2B \land \neg C1B$

b. **Prove that box 2 must contain white balls (C2W) using Natural Deduction.**

Prove KB |= C2W by Natural Deduction:

1. From O3Y and O3Y→C3Y∨C3B
    a. Derive C3Y∨C3B by MP
2. From L3B and L3B→¬C3B
    a. Derive ¬C3B by MP
3. From C3Y∨C3B and ¬C3B
    a. Derive C3Y by Resolution
4. From C3Y and C3Y→¬C2Y∧¬C1Y
    a. Derive ¬C2Y∧¬C1Y by MP
5. From O1Y and O1Y→C1Y∨C1B
    a. Derive C1Y∨C1B by MP
6. From ¬C2Y∧¬C1Y
    a. Derive ¬C1Y by AE
7. From C1Y∨C1B and ¬C1Y
    a. Derive C1B by Resolution
8. From O2W and O2W→C2W∨C2B
    a. Derive C2W∨C2B by MP
9. From C1B and C1B→¬C2B∧¬C3B
    a. Derive ¬C2B∧¬C3B
10. From ¬C2B∧¬C3B
    a. Derive ¬C2B
11. From C2W∨C2B and ¬C2B
    a. Derive C2W by Resolution

c. **Convert your KB to CNF.**

Most of the statements in the KB are not used for part d, but here they are in CNF:

1. O1W → C1W ⋁ C1B
    a. ¬O1W ⋁ C1W ⋁ C1B
2. O2Y → C2Y ⋁ C2B
    a. ¬O2Y ⋁ C2Y ⋁ C2B
3. O3W → C3W ⋁ C3B
    a. ¬O3W ⋁ C3W ⋁ C3B
4. L1Y → ¬C1Y
    a. ¬L1Y ⋁ ¬C1Y
5. L1W → ¬C1W
    a. ¬L1W ⋁ ¬C1W
6. L1B → ¬C1B
    a. ¬L1B ⋁ ¬C1B
7. L2Y → ¬C2Y
    a. ¬L2Y ⋁ ¬C2Y
8. L2W → ¬C2W
    a. ¬L2W ⋁ ¬C2W
9. L2B → ¬C2B
    a. ¬L2B ⋁ ¬C2B
10. L3Y → ¬C3Y
    a. ¬L3Y ⋁ ¬C3Y
11. L3W → ¬C3W
    a. ¬L3W ⋁ ¬C3W
12. C1Y ⋁ C1W ⋁ C1B
13. C2Y ⋁ C2W ⋁ C2B
14. C3Y ⋁ C3W ⋁ C3B
15. C1Y → ¬C2Y ⋀ ¬C3Y
    a. ¬C1Y ⋁ ¬C2Y, ¬C1Y ⋁ ¬C3Y
16. C1W → ¬C2W ⋀ ¬C3W
    a. ¬C1W ⋁ ¬C2W, ¬C1W ⋁ ¬C3W
17. C2Y → ¬C1Y ⋀ ¬C3Y
    a. ¬C2Y ⋁ ¬C1Y, ¬C2Y ⋁ ¬C3Y
18. C2W → ¬C1W ⋀ ¬C3W
    a. ¬C2W ⋁ ¬C1W, ¬C2W ⋁ ¬C3W
19. C2B → ¬C1B ⋀ ¬C3B
    a. ¬C2B ⋁ ¬C1B, ¬C2B ⋁ ¬C3B
20. C3W → ¬C2W ⋀ ¬C1W
    a. ¬C3W ⋁ ¬C2W, ¬C3W ⋁ ¬C1W
21. C3B → ¬C2B ⋀ ¬C1B
    a. ¬C3B ⋁ ¬C2B, ¬C3B ⋁ ¬C1B

Only the statements below from the KB, when converted to CNF, are used for part d:

1. O3Y→C3Y⋁C3B

    a. $\neg O3Y \lor C3Y \lor C3B$

2. $O1Y \to C1Y \lor C1B$
    a. $\neg O1Y \lor C1Y \lor C1B$
3. $O2W \to C2W \lor C2B$
    a. $\neg O2W \lor C2W \lor C2B$
4. $L3B \to \neg C3B$
    a. $\neg L3B \lor \neg C3B$
5. $C1B \to \neg C2B \land \neg C3B$
    a. $\neg C1B \lor \neg C2B$
    b. $\neg C1B \lor \neg C3B$
6. $C3Y \to \neg C2Y \land \neg C1Y$
    a. $\neg C3Y \lor \neg C2Y$
    b. $\neg C3Y \lor \neg C1Y$

d. **Prove C2W using Resolution.**

First we represent our facts as unit clauses:

7. O1Y
8. O2W
9. O3Y
10. L1W
11. L2Y
12. L3B

Then we negate the query and proceed:

13. ¬C2W
14. From O3Y and ¬O3Y∨C3Y∨C3B
    a. Derive C3Y∨C3B by Resolution
15. From L3B and ¬L3B∨¬C3B
    a. Derive ¬C3B by Resolution
16. From C3Y∨C3B and ¬C3B
    a. Derive C3Y by Resolution
17. From C3Y and ¬C3Y∨¬C1Y
    a. Derive ¬C1Y by Resolution
18. From O1Y and ¬O1Y∨C1Y∨C1B
    a. Derive C1Y∨C1B by Resolution
19. From ¬C1Y and C1Y∨C1B
    a. Derive C1B by Resolution
20. From C1B and ¬C1B∨¬C2B
    a. Derive ¬C2B by Resolution
21. From O2W and ¬O2W∨C2W∨C2B
    a. Derive C2W∨C2B by Resolution
22. From ¬C2B and C2W∨C2B
    a. Derive C2W by Resolution
23. From C2W and ¬C2W
    a. [] - empty clause - Proof by Contradiction is done

3. Do Forward Chaining for the CanGetToWork KB below. You don't need to follow the formal FC algorithm (with agenda/queue and counts array). Just indicate which rules are triggered (in any order), and keep going until all consequences are generated. Show the final list of all inferred propositions at the end. Is CanGetToWork among them?

```
KB = { a. CanBikeToWork → CanGetToWork
       b. CanDriveToWork → CanGetToWork
       c. CanWalkToWork → CanGetToWork
       d. HaveBike ∧ WorkCloseToHome ^ Sunny → CanBikeToWork
       e. HaveMountainBike → HaveBike
       f. HaveTenSpeed → HaveBike
       g. OwnCar → CanDriveToWork
       h. OwnCar → MustGetAnnualInspection
       i. OwnCar → MustHaveValidLicense
       j. CanRentCar → CanDriveToWork
       k. HaveMoney ∧ CarRentalOpen → CanRentCar
       l. HertzOpen→ CarRentalOpen
       m. AvisOpen→ CarRentalOpen
       n. EnterpriseOpen→ CarRentalOpen
       o. CarRentalOpen → IsNotAHoliday
       p. HaveMoney ∧ TaxiAvailable → CanDriveToWork
       q. Sunny ^ WorkCloseToHome → CanWalkToWork
       r. HaveUmbrella ^ WorkCloseToHome → CanWalkToWork
       s. Sunny → StreetsDry }
```

Facts: { `Rainy, HaveMoutainBike, EnjoyPlayingSoccer, WorkForUniversity, WorkCloseToHome, HaveMoney, HertzClosed, AvisOpen, McDonaldsOpen` }

1. Rainy fact does not trigger anything
2. HaveMountainBike fact triggers rule e
    a. Now we get the inferred proposition HaveBike
        i. HaveBike does not trigger anything
3. EnjoyPlayingSoccer fact does not trigger anything
4. WorkForUniversity fact does not trigger anything
5. WorkCloseToHome fact does not trigger anything
6. HaveMoney fact does not trigger anything
7. HertzClosed fact does not trigger anything
8. AvisOpen fact triggers rule m
    a. Now we get the inferred proposition CarRentalOpen
        i. We now have CarRentalOpen and HaveMoney, so we trigger rule k
            1. Now we get the inferred proposition CanRentCar
                a. CanRentCar triggers rule j
                    i. Now we have the inferred proposition CanDriveToWork
                    ii. CanDriveToWork triggers rule b
                        1. Now we have the inferred proposition CanGetToWork

4. **Do Backward Chaining for the CanGetToWork KB.**

In this case, you should follow the BC algorithm closely (the pseudocode for the propositional version of Back-chaining is given in the lecture slides).

Important: when you pop a subgoal (proposition) from the goal stack, you should systematically go through all rules that can be used to prove it IN THE ORDER THEY APPEAR IN THE KB. In some cases, this will lead to *back-tracking*, which you should show.

Also, the sequence of results depends on order in which antecedents are pushed onto the stack. If you have a rule like $A \wedge B \rightarrow C$, and you pop C off the stack, push the antecedents in reverse order, so B goes in first, then A; in the next iteration, A would be the next subgoal popped off the stack.

1. {CanGetToWork}
    a. Initialize goal stack with query
2. {CanBikeToWork}
    a. Replace goal with rule a
3. {HaveBike, WorkCloseToHome, Sunny}
    a. Push antecedents for rule d
4. {HaveMountainBike, WorkCloseToHome, Sunny}
    a. Push antecedents for rule e
5. {WorkCloseToHome, Sunny}
    a. HaveMountainBike is a fact, so pop it
6. {Sunny}
    a. WorkCloseToHome is a fact, so pop it
7. Backtrack - Sunny is not a fact, nor can it be proved
8. {CanDriveToWork}
    a. Replace goal with rule b
9. {OwnCar}
    a. Push antecedents for rule g
10. Backtrack - OwnCar is not a fact, nor can it be proved
11. {CanRentCar}
    a. Push antecedents for rule j
12. {HaveMoney, CarRentalOpen}
    a. Push antecedents for rule k
13. {CarRentalOpen}
    a. HaveMoney is a fact, so pop it
14. {HertzOpen}
    a. Push antecedents for rule l
15. Backtrack - HertzOpen is not a fact, nor can it be proved
16. {AvisOpen}
    a. Push antecedents for rule m
17. {}
    a. AvisOpen is a fact, so pop it
    b. Success! The goal stack is empty, so return True