Exercises on Logical Inference - CSE 473 – Spring 2006

You should do these exercises, but they will not be turned in or graded. Instead, a solution sheet will be given out so you can check your own work. **You are responsible for understanding how to solve problems like these, and similar questions may appear on the final exam.**

(1) Translate the following sentences into propositional logic. Specify a single set of simple propositions that are used by the sentences, and indicate the intended meaning of each. After translating each directly into logic convert the formulas into conjunctive normal form.

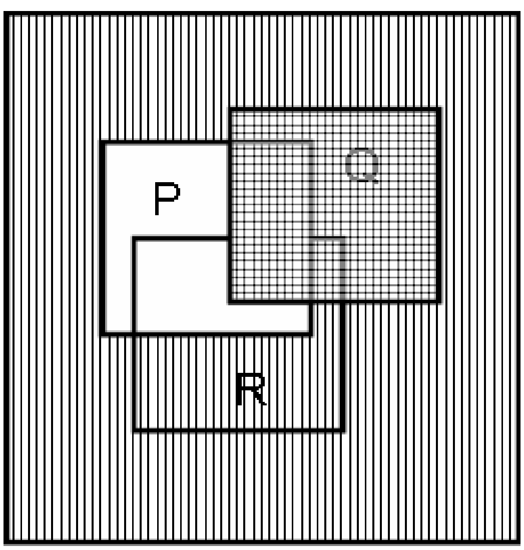
(1.1) It will snow tomorrow, and if the kumquats are in blossom the crop will be ruined.

(1.2.) If it snows tomorrow and the kumquats are in blossom, then the crop will be ruined.

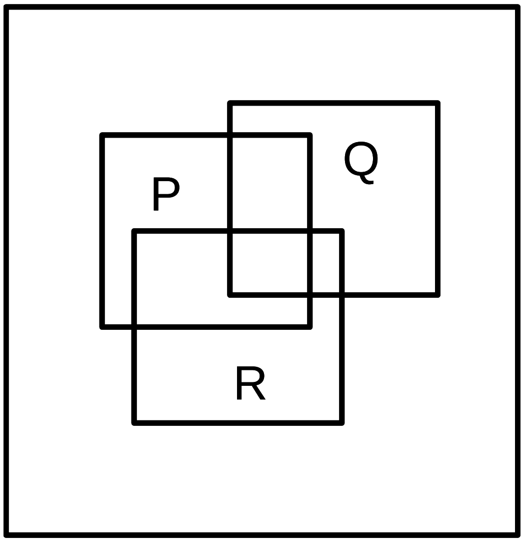
(1.2) If it snows or freezes tomorrow, then if the kumquats are in blossom and are unprotected, then the crop will be ruined unless a miracle occurs.

(2) For each of the following pairs of formulas shade the area of the Venn diagram that contains truth-assignments that make formula "A" true with horizontal lines, and shade the area that makes formula "B" true with vertical lines. Then indicate whether or not formula A entails formula B.

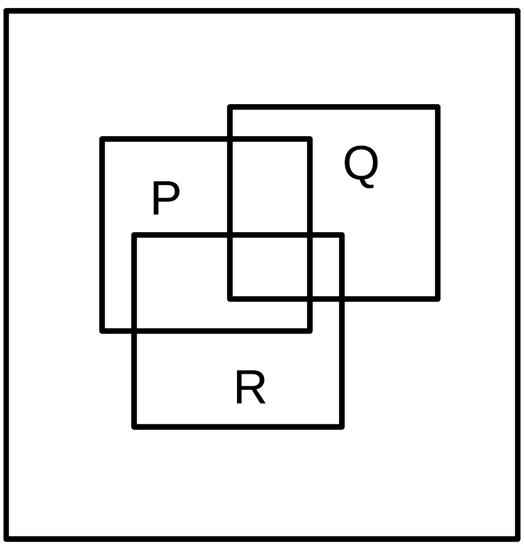
Example: [A] Q [B] $P \supset Q$ Does [A] entail [B]? YES



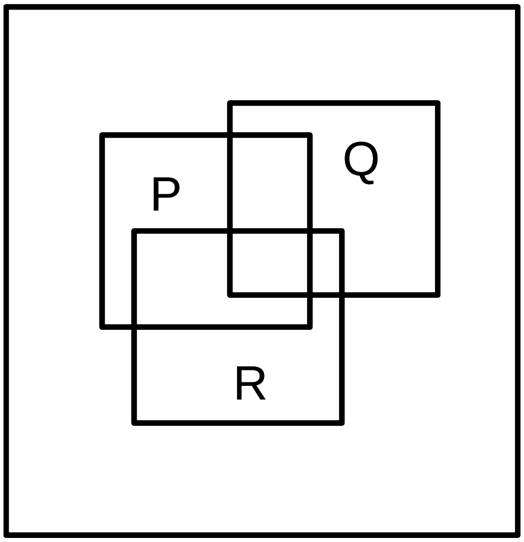
(2.1) [A] $P \supset Q$ [B] $P \supset (Q \vee R)$ Does [A] entail [B]?



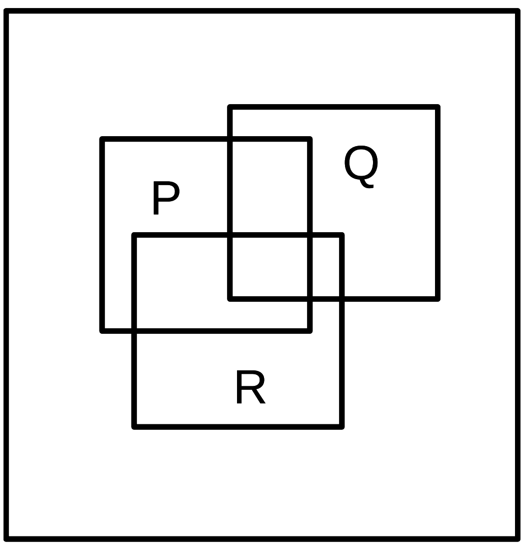
(2.2) [A] $(P \wedge Q) \supset R$ [B] $P \supset R$ Does [A] entail [B]?



(2.3) [A] $(P \vee Q) \supset R$ [B] $P \supset R$ Does [A] entail [B]?



(2.4) [A] $(P \supset Q) \wedge (Q \supset R)$ [B] $P \supset R$ Does [A] entail [B]?

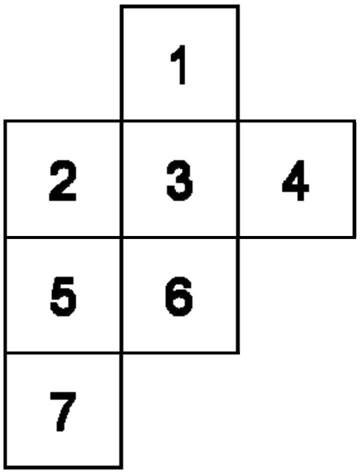


(3) A crossword puzzle can be represented in propositional logic so that each model of the formula corresponds to a solution to the puzzle. This can be done as follows:

a) A proposition is introduced for each square of the puzzle and each letter of the alphabet. Clauses are added that assert that no square is filled in with two different letters.
b) A proposition is introduced for each word of in the dictionary of length k and each position in the puzzle for words of length k. A position is a pair of a start and end square. Clauses are added that assert that each position in the puzzle contains a word of the proper length.
c) Clauses are added that assert that if a word occurs at a position, all the letters in the word fill the corresponding squares in the puzzle.

Below is a tiny fragment of a crossword puzzle and dictionary. We supply the clauses of types (a) and (b). Your job is to write down the clauses of type (c) for this specific fragment and dictionary, as specified below. You should also state the total number of type (c) clauses for this fragment. You should use conjunction normal form (CNF). You should not use any quantifiers or notation other than plain CNF.

Puzzle fragment:



Dictionary: at cat con on pan son

Type (a) clauses:

$\neg fills(1,a) \vee \neg fills(1,b)$

$\neg fills(1,a) \vee \neg fills(1,c)$

...

$\neg fills(7,y) \vee \neg fills(7,z)$

Type (b) clauses:

$position(1,6,son) \vee position(1,6,pan) \vee position(1,6,can) \vee position(1,6,con)$

$position(2,4,son) \vee position(2,4,pan) \vee position(2,4,can) \vee position(2,4,con)$

$position(2,7,son) \vee position(2,7,pan) \vee position(2,7,can) \vee position(2,7,con)$

$position(5,6,at) \vee position(5,6,on)$

(4) The logical encoding of many problems can be specified in a compact form using *clause schemas*, where each *schema* is generates a large number of propositional clauses. We can think of a schema as a little program in pseudo-code that generates a set of propositional clauses when it is executed. For example, consider the 8-Queens problem. The clauses that assert that there is exactly one queen in each row can be written compactly using the following schemas:

for (i=1; i ≤ 8; i++) { (Q(i,1) ∨ Q(i,2) ∨ Q(i,3) ∨ Q(i,4) ∨ Q(i,5) ∨ Q(i,6) ∨ Q(i,7) ∨ Q(i,8)) }

```
for (i=1; i ≤ 8; i++)
        for (j=1; j ≤ 8; j++)
                for (k=1; k ≤ 8; k++)
                        if (j≠k) { (¬Q(i,j) ∨ ¬Q(i,k)) }
```

Write clause schemas to generate the clauses that assert that there are no attacks between queens in the same column.

(5) The Walksat procedure tries to flip variables that would decrease the number of unsatisfied clauses in the formula. It performs badly on formulas where the local neighborhood of an assignment provides no guidance about which variable to flip. An example of such a “bad” formula is one that has a single solution, and furthermore every non-satisfying truth assignment violates exactly one clause. (Not the same clause – one clause total!) Write down this CNF formula for the case of three variables, A, B and C. Prove that your answer is correct by indicating which clause is false on each non-solution truth assignment. Hint: think about how many truth assignments a clause of a given length rules out.