CS 153 **Foundations of Software** **Homework 5**
Fall 2006 **Due: Nov 3, 2006**

1. [5] Exercise 6.2.1 (p. 367)

2. [5] Use truth tables to verify the equivalence $A \lor (A \land B) \;\equiv\; A$.

3. [10] Exercise 6.2.8(b) (p. 367)

4. [10] Exercise 6.2.9(d) (p. 368)

5. [10] Exercise 6.2.10(d) (p. 368)

6. [10] Exercise 6.2.11(f) (p. 368)

7. [10] Exercise 6.2.12(f) (p. 368)

8. [40] Write (and test and debug) a program in your favorite language that will evaluate the truth of a WFF given a truth assignment (or interpretation) for its propositional variables. You will need to do the following:

(a) Define a data structure representing WFFs, including propositional variables. You might, for instance, represent propositional variables as strings, or possibly as numbers (integers, say).

(b) Define a data structure or other representation for truth assignments. You could, for instance, use an *association list* data structure consisting of a list or sequence of ordered pairs, where each ordered pair would consist of a propositional variable and its assigned truth value. E.g. the assignment

$$\{\mathsf{P} \mapsto \mathsf{T}, \mathsf{Q} \mapsto \mathsf{F}, \mathsf{R} \mapsto \mathsf{T}\}$$

would be represented by the association list (using ML list syntax):

$$[(\mathsf{P}, \mathsf{T}), (\mathsf{Q}, \mathsf{F}), (\mathsf{R}, \mathsf{T})]$$

(c) Define an evaluator that takes as parameters a WFF and a truth assignment, and returns true or false as appropriate.