CS 153	**Foundations of Software**	**Homework 7**
Fall 2006		**Due: Nov 20, 2006**

1. [5] Exercise 7.1.1(b) (p. 413) [Hint: Use answer to part (a) as a model of what is expected.]

2. [5] Exercise 7.1.5(b) (p. 413)

3. [5] Exercise 7.1.7(b) (p. 414)

4. [5] Exercise 7.1.9(b) (p. 415)

5. [10] Exercise 7.1.10(b) (p. 415)

6. [10] Exercise 7.1.11(b) (p. 415)

7. [10] Exercise 7.1.12(d) (p. 415)

8. [10] Exercise 7.1.14(b) (p. 415)

9. [5] Exercise 7.1.16(b) (p. 416)

10. [5] Exercise 7.1.17(b) (p. 415)

11. [30] Write (and test and debug) a program in Standard ML (SML) that represents terms and formulas of a predicate calculus over a language that has the function symbols e (nullary) and $*$ (binary) and an relation symbol *eq* (these function symbols might represent the identity element and multiplication operator of a *monoid* – look it up in Wikipedia if you are not familiar with monoids). You can use the supplied file pc.sml as a starting place (see the class web page).

(a) Write a function that calculates the set of free variables of a formula. Then use this to write functions that produce the universal and existential closures of a given formula.

(b) Write a function that implements substitution of a term for free occurrences of a variable in a formula (`subst(A,x,t)`=$A[t/x]$). What can you do about situations like the following? $(\exists x\, p(x,y))[f(x)/y] =_? \exists x\, p(x, f(x))$ (an instance of undesirable *free variable capture*).