NAME (PRINT):			
	Last/Surname	First /Given Name	
STUDENT #:		**SIGNATURE:**	

UNIVERSITY OF TORONTO MISSISSAUGA
DECEMBER 2013 FINAL EXAMINATION
CSC324H5F
Principles of Programming Languages
Anthony Bonner
Duration - 2 hours
Aids: Open book (Textbook)

The University of Toronto Mississauga and you, as a student, share a commitment to academic integrity. You are reminded that you may be charged with an academic offence for possessing any unauthorized aids during the writing of an exam. Clear, sealable, plastic bags have been provided for all electronic devices with storage, including but not limited to: cell phones, tablets, laptops, calculators, and MP3 players. Please turn off all devices, seal them in the bag provided, and place the bag under your desk for the duration of the examination. You will not be able to touch the bag or its contents until the exam is over.

If, during an exam, any of these items are found on your person or in the area of your desk other than in the clear, sealable, plastic bag; you may be charged with an academic offence. A typical penalty for an academic offence may cause you to fail the course.

*Please note, you **CANNOT** petition to **re-write** an examination once the exam has begun.*

Write your answers on the examination sheet in the spaces provided. You may use the backs of pages if necessary. Concise, well-written answers will receive more points than long, rambling ones. Unless stated otherwise, all answers should be justified.

If you do not know the answer to a question, and you write "I don't know", you will receive 20% of the marks of that question. If you just leave a question blank with no such statement, you get 0 marks for that question.

This page is for marking purposes only

Question	Value	Score
1	28	
2	38	
3	17	
4	15	
5	26	
6	25	
Total:	149	

1. **Short Answers: 28 marks total.**

   (a) (4 marks) A programmer wishes to write some facts in Prolog stating that all birds can fly, except for ostriches and penguins. To do this, he writes the following:

   ```
   notFlies(Bird) :- ostrich(Bird).
   notFlies(Bird) :- penguin(Bird).
   flies(Otherbirds).
   ```

   Explain why this doesn't do what is wanted, and provide Prolog code that corrects the problem.

   (b) (2 marks) What is the difference between the following two Prolog rules?

   ```
   p(U) :- q(V,U), r(V).
   ```

   ```
   p(U) :- q(_,U), r(_).
   ```

   (c) (4 marks) For each of the following pairs, state whether or not the atomic formulas unify. In each case, state what the most general unifier is, or show that unification is impossible. Here, U, V, W, X, Y, and Z are variables, and a is a constant symbol.

continued on page 3

i. $p(V, g(f(X), V))$ and $p(h(U), g(W, W))$

ii. $q(Y, f(Y), h(a))$ and $q(g(Z), X, Z)$.

(d) (4 marks) Consider the following ML function:

```
fun f [] = 0.0
  | f ((X,Y)::L) = X*Y + f(L);
```

i. In plain English, what does this function do?

ii. What is the type of this function?

continued on page 4

(e) (6 marks) Consider the following Scheme function:

```
(define (f u v w) (lambda (x) (u v (u w x))))
```

i. What is the value of the following expression:

```
((f list 47 13) 26)
```

ii. Redefine this function in ML.

iii. What is the type of this ML function?

(f) (8 marks) Suppose that the predicate $flight(N, C1, C2)$ means that flight N goes from city $C1$ to city $C2$. Using only this predicate, write formulas in first-order predicate logic that express the ideas below. You may use non-Horn formulas and constant symbols, but do not use any function symbols.

i. There is a flight from Boston to Chicago.

continued on page 5

ii. There are no flights into Bloomington

iii. AC309 is the one and only flight into Barrie.

iv. AC517 and WJ2234 are the only flights out of Moosejaw

continued on page 6

2. **ML: 38 marks total.**

   In this question, your ML definitions should use pattern matching whenever possible. Simple functions are preferred to complex ones. You may define helper functions.

   We define a blue-green tree as follows: each node is coloured either blue or green; a blue node has at most two children, while a green node has at most three children; a blue node stores an integer, while a green node stores references to two real numbers.

   (a) (4 marks) Define an ML datatype called `bgTree` for blue-green trees.

continued on page 7

(b) (4 marks) Draw a picture of a blue-green tree, and show how it is represented in your ML datatype. Choose a tree that has at least one of each of the following: a blue node with two children, a green node with three children, a blue node with no children, a green node with one child, and a green node with no children.

continued on page 8

(c) (15 marks) Write an ML function `blueMax(T)` of type `bgTree -> int` that returns the maximum of all the integers stored in the blue nodes of a blue-green tree, `T`. In this question, the blue nodes are assumed to store store only non-negative numbers; so, raise an exception if `T` contains any blue nodes with negative numbers. If `T` contains no blue nodes, return 0.

continued on page 9

(d) (15 marks) Write an ML function `greenAdd1(T)` of type `bgTree -> unit` that updates a blue-green tree, `T`, by adding 1.0 to all the real numbers stored in its green nodes. All updates should be done using assignment statements, not by creating a modified copy of `T`.

continued on page 10

3. **Proving Properties of Programs: 17 marks total.**

Consider the following two Scheme functions:

```
(define (f X)
    (if (null? X) X (cons (* 2 (car X)) (f (cdr X)))))

(define (g Y Z)
   (if (null? Y) Z
       (cons (car Y) (g (cdr Y) Z))))
```

(a) (4 marks) Write down the basic properties of these functions (*i.e.*, those properties needed to answer part (b) of this question).

continued on page 11

(b) (13 points) Use induction to prove that these functions have the following property:

```
(f (g X Y)) = (g (f X) (f Y))
```

continued on page 12

4. **Scheme: 15 marks.**

   Suppose `P` and `Q` are unary functions. Define a Scheme function `(replace P Q E)` that replaces `z` by `(Q z)` for every symbol or number `z` in s-expression `E` for which `(P z)` is true. For example,

```
(replace number? square '(2 a 3 b))  =>  (4 a 9 b)

(replace number? square '(2 (a (3 b))))  =>  (4 (a (9 b)))

(replace number? even? '(a (b 3) 4))  =>  (a (b #f) #t)

(replace even? add1 '((1 2) (3 4)))  =>  ((1 3) (3 5))

(replace even? add1 8)  =>  9

(replace symbol? list '(a 1 (b 3) c 6))  =>  ((a) 1 ((b) 3) (c) 6)
```

continued on page 13

5. **Prolog Databases: 26 marks total.**

Suppose you are given a Prolog database of Toronto banks, bank accounts, and people. The database consists of ground atomic formulas made from the following predicates:

```
bank(bank_name,branch,manager,street,number)
account(bank_name,branch,number,balance,person)
works(person,bank_name,branch,salary)
lives(person,street,number)
```

where the attributes are as follows:

- `bank_name` is the name of a bank, *eg*, `cibc`, `td`, `scotiabank`, `royalbank`, etc.
- `branch` is a constant identifying a branch of a bank, usually by describing its rough location within Toronto, *eg*, `downtown`, `westside`, `harbourfront`, `yonge-and-bloor`, `bloor-west-village`, etc.
- `manager` is the name of the manager of a particular bank branch.
- `person` is the name of a person.

Here are typical atomic formulas for each of the four predicates:

```
bank(cibc,westside,scrooge,danforth,1313)
account(cibc,westside,456,725000,yeltsin)
works(krachit,cibc,westside,25000)
lives(krachit,steeles,5765)
```

- The first formula means that the westside branch of CIBC is managed by Scrooge, and is located at 1313 Danforth Ave.
- The second formula means that at the westside branch of CIBC, account number 456 has a balance of $725,000, and is held by Yeltsin.
- The third formula means that Krachit works at the westside branch of CIBC, and earns a salary of $25,000 a year.
- The fourth formula means that Krachit lives at 5765 Steeles Ave.

With this in mind, answer the questions below. Your Prolog code should follow good logic-programming style, as required in the homework assignments.

continued on page 14

(a) **Database Queries: 12 marks total**

Without adding any rules to the Prolog database, pose queries to Prolog that retrieve the the information below (and no more). If necessary, you may use inequality predicates; e.g., $x < y$ is true iff x is less than y. Recall that values bound to the underscore variable (_) are not printed.

i. (2 marks) The names of all people working for CIBC.

ii. (3 marks) The name, home address and account balance of everyone with an account balance of over $100,000.

iii. (3 marks)
The name and home address of all CIBC bank managers.

iv. (4 marks) The employee name, home address, bank branch and bank address of all CIBC employees.

continued on page 15

(b) **Deductive Databases: 14 marks total**

Add rules to the Prolog database that define the predicates below.

i. (4 marks) `empMan(E,M)`, which means that `E` is the name of an employee and `M` is the name of his manager (*i.e.*, the manager of the bank branch where the employee works).

ii. (4 marks) `streetPerson(P,S)`, which means that `P` is the name of a person who has an account at a bank branch located on street `S`.

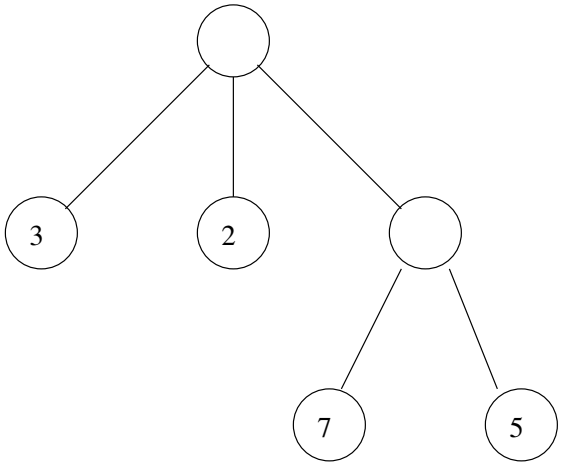
iii. (6 marks) `twoAccounts(Name,Street,Number)`: the name and address of a person who has at least two different bank accounts at the same bank branch.

continued on page 16

6. **Logic Programming: 25 marks total.**

We shall use function terms and lists to represent trees. In these trees, each internal node has a list of children, and each leaf node stores a number. Specifically, the function term `intNode(L)` represents an internal node, where `L` is a non-empty list of children. Likewise, the function term `leafNode(X)` represents a leaf node, where `X` is a number. In the questions below, your Prolog code should follow good logic-programming style, as required in the homework assignments.

(a) (3 marks) Using this representation, how is the tree below represented as a function term?



continued on page 17

(b) (4 marks) Define a Prolog predicate `leftMost(T,N)` that is true iff `N` is the number stored in the left-most leaf of tree `T`. Thus, if `T` represents the tree drawn in part (a) above, then `N=3`.

(c) (8 marks) Define a Prolog predicate `rightMost(T,N)` that is true iff `N` is the number stored in the right-most leaf of tree `T`. Thus, if `T` represents the tree drawn in part (a) above, then `N=5`. For full marks, your solution should use mutual recursion.

continued on page 18

(d) (10 marks) Define a Prolog predicate `squarePos(T1,T2)` that is true iff tree `T2` is identical to tree `T1` except that every positive number in `T1` has been squared (multiplied by itself). All other numbers must remain the same. For full marks, your solution should use mutual recursion. You may use the inequality predicate , $X > Y$.

continued on page 19

Use this page for rough work.

continued on page 20

Use this page for rough work

Total marks = 149 **END OF EXAM**