Brock University
Instructor: Brian Ross

NAME (print): _____

STUDENT NUMBER: _____

There are **5** questions totalling **51** marks. There are **6** pages in the exam.
Please answer all questions on the exam paper. Use the backs of pages if necessary.

_____

Question 1 [14]

a) [6] Identify and briefly discuss the 3 kinds of clauses used in Prolog progams

b) [4] Define the term "predicate". What is the relationship between clauses and predicates?

c) [4] Briefly discuss the two essential characteristics of recursive programs.

Question 2 [7] For each expression given, indicate the final result when interpreted. If it fails, explain why.

(a) ?- Six is 6, Seven = 7, Eight is Six+Seven.

(b) ?- struct(Donald, Trump, [is, rich]) = DontFireMe.

(c) ?- [a, b, c, d] = [Donald, _, Trump | Thing].

(d) ?- Paris = Hilton, s(a, Hilton) = s(A, _, Hilton), X is 1+1.

(e) ?- CarForBrian is mazda(RX8, please).

(f) ?- Spam = [spam | Spam].

(g) ?- unusualTerm(2.5, [a], paris_hilton) = unusualTerm(Num, List, brian_ross).

Question 2 [2+2+4 = 8]

(a) Consider the following:

```
member(A, [A|T]).                  %1
member(A, [B|T]) :- member(A, T).  %2
```

How many clauses are in the above: _____

How many predicates: _____

Which variables are "singleton": _____

Which variable(s) can be replaced with an anonymous variable: _____

(b) [2] What does member/2 do if the rule is **replaced** with the following:

```
member(A, [A|T]) :- member(A, T).
```


(b) [4] Draw a complete computation tree for the query: **?- member(a, [a, b]).**

Question 4 [4+6 = 10]

(a) Write a predicate last/2. It takes a list, and returns the last element in it. For example,

```
?- last([a,b,c], L).
L = c
?- last([ ], L).
no
```

(b) Write a predicate count/3. It takes a list and constant, and counts the number of times the constant is in the list. For example:

```
?- count([a,b,a,c], a, N).
N = 2
?- count([a,b,a,c], d, N).
N = 0
```

Question 5 [4+8 = 12]

a) Write a predicate list/1. It succeeds if its argument is a list. Otherwise it fails.

b) Write a predicate replaceList/2. It takes a list, and creates a new list as follows. In the new list, every number in the list is replaced by the constant 'number'. Similarly every list is replaced by 'list', and every other constant is replaced by 'constant'. Variables are left as they are. For example:

> ?- replaceList([a, [ ], 3, b, 4.2, [25, 7], X, d], Ans).
> Ans = [constant, list, number, constant, list, X, constant].

You may use the builtins number/1, atom/1, and var/1, which succeed if the arguments are numbers, constants, or variables.

*** The End ***