Name:____

SSN:____

0	1	2	3	4	5	6	7	8	9	10	total
0	*20*	*40*	*15*	*20*	*40*	*20*	*25*	*15*	*15*	*30*	*240*

UMBC CMSC471 Final Exam
December 19, 2001

If a thing is worth doing, it's worth doing well - unless doing it well takes so long that it isn't worth doing any more. Then you just do it 'good enough'.
-- Programming Perl, Wall and Schwartz

Please write all of your answers on this exam. The exam is closed book and consists of 10 problems which add up to 240 points. You have two hours to work on this exam. Good luck.

0. Warm Up Exercise (0)

There are fifteen birds on a tree. You shoot one bird, and it falls down to the ground dead. How many are left on the tree? (a) 15 (b) 14 (c) 0

1. Unification (20)

Each of the following Prolog queries invokes unification. For each, say whether or not the unification would succeed and, if it does, what values the variables in the terms would take on. Assume that the same variable name in two different terms represents the same variable.

1. loves(John,John)=loves(John,mary).
2. f(X)=f(f(X)).
3. p(X,q(Y),r(Z)) = p(r(Y), q(Z), r(X)).
4. X+Y='+'(1,2).
5. nil=[].
6. [[]]=[X|X].
7. [[[]],[[]],[]]=[[X]],[X],X].
8. f(g(X),Y,Z)=f(Y,g(X),g(Y)).
9. (Foo:-Bar) = (Bar -> Foo).
10. (X+(Y+Z))=((X+Y)+Z).

2. True/False (40 points)

Circle either T or an F in the space before each statement to indicate whether the statement is true or false. If you think the answer is simultaneously true and false, quit while you are ahead.

T F A sound inference system is able to derive all of the conclusions from an initial KB that logically follow from it.

T F A complete inference system is one which supports all of the basic logical connectives (e.g., and, or, not, implies and iff).

T F A valid sentence is one which is true under all interpretations.

T F In a monotonic logic, adding a new sentence to a knowledge base KB1 produces a knowledge base KB2 in which all of the sentences that were in KB1 are also in KB2

T F First order predicate logic extends propositional logic by adding variables and quantifiers.

T F It is possible to translate every sentence in first order logic into a set of sentences in conjunctive normal form.

T F Conjunctive normal form is a representation in which the KB is a conjunction of clauses where each clause is a conjunction of literals.

T F In first order logic, a sentence is satisfiable if and only if all if its variables are within the scope of either a universal or an existential quantifier.

T F Every legal sentence in propositional logic is also a legal sentence in first order logic.

T F While both forward chaining and backward chaining are sound inference techniques, only backward chaining is complete.

T F A partial order planning system is less powerful than a strips-like planning system because it can only derive partial plans.

T F Any sentence in a first order knowledge base KB that is not entailed by KB can be proven false by a resolution theorem prover.

T F Given two clauses in a first order KB, there may be more than one way to unify them.

T F Given two clauses in a first order KB, there may be more than one way to resolve them.

T F The ID3 algorithm for decision tree induction (from which C5 is descended) will always produce the smallest possible tree, as measured in the number of non-leaf nodes.

T F Decision tree learning can only be used to predict the value of a binary attribute, i.e. one with only two possible values.

T F Supervised learning can only be done with a set of training examples for which the "right answer" is known.

T F Using the alpha-beta algorithm in the prisoner's dilemma would be a good idea.

T F The mini-max and alpha-beta procedures will **always** back up identical values to the root node of a game tree.

T F The study of emergent behavior from systems of local rules in the simulations of ants and flocks of birds just shows how truly dumb bugs and birds are.

3. Prolog (15)

Write a Prolog program called separate(List, Numbers, NonNumbers) that takes a simple list of terms (List), and puts all of the numbers in one list (Numbers) and all of the non numeric items in another list (NonNumbers). You can use the number/1 predicate to check if something is a number. Assume that the first argument is always instantiated. Your program should work for the following query:

```
?- separate([a,b,4,1,[1],4.2E+01],Nums,Others).
     Nums = [4,1,4.2E+01]
     Others = [a,b,[1]]
```

4. Clausal form (20)

Convert each of the following sentences into clausal normal form and implicative normal form.

(a) $\neg (\forall x\ \forall y\ (p(X,Y) \land \neg\ q(f(X),g(X,Y))))$

(b) $\exists x\ \forall y\ \exists z\ (p(x,y,z) \Leftrightarrow p(z,y,x))$

5. English to Logic (40 points)

Encode each the following English sentences as a single sentence in first order logic. Use the predicates yellow/1, red/1, apple/1, fruit/1, delicious/1 and moreDelicious/2.

(a) All yellow apples are delicious.

(b) All delicious apples are yellow.

(c) Some delicious things are yellow apples.

(d) Every apple is either yellow or red.

(e) No apple is both yellow and red.

(f) An apple is a fruit.

(g) Not all fruits are apples.

(h) The most delicious fruit is an apple.

(i) Yellow apples are more delicious than red apples.

(j) Some red fruit is more delicious than some yellow fruit.

6. Logic to English (20)

Give a simple, unambiguous English sentence that is a reasonable translation for each of the following sentences in first order logic sentences. Make reasonable guesses about the meaning of the predicates used.

(a) $\forall X\ person(X) \Rightarrow male(X) \vee female(X)$

(b) $\forall X\ boy(X) \Leftrightarrow male(X) \wedge young(X)$

(c) $\neg\ (\exists X\ male(X) \wedge female(X))$

(d) $\exists X\ male(X) \wedge \neg\ person(X)$

(e) $\forall X\ person(X) \wedge parent(X,Y) \Rightarrow\ person(Y)$

7. Theorem proving (25)

Consider the following formalization of the facts in a Retriever article. Use resolution theorem proving to show that the sentences are inconsistent. Start by putting the sentences into clausal form, then show that false can be derived from them thru repeated application of the resolution rule of inference.

UMBC student Alice said that UMBC should serve beer in the Pub.
$\quad$ student(alice) $\wedge$ (serveBeer(umbc,pub) $\Rightarrow$ listento(umbc,alice))
UMBC student Bill said that UMBC shouldn't serve beer in the Pub.
$\quad$ student(bill) $\wedge$ ($\neg$ serveBeer(umbc,pub) $\Rightarrow$ listento(umbc,bill))
UMBC student Carol said that UMBC never listens to students.
$\quad$ $\forall$x (student(X) $\Rightarrow$ $\neg$ listento(umbc,X))

8. Strips planning (15)

A friend says that the blocks world model could be greatly simplified by having a simple *on(X,Y)* predicate that says that X is on top of Y where Y can either be another block or the table. She claims that we can now replace the four operators *stack*, *unstack*, *pickup* and *putdown* with the following *move* operator. Will her suggestion work? Explain why it will or why it won't.

```
operator(move(B,X,Y),             %  move block B from being on X to on Y.
   [on(B,X),clear(B),clear(Y)],   %  preconditions: B's on X and both B and Y are clear
   [on(B,Y),clear(X)],            %  add list: B is now on Y and X is clear
   [clear(Y),on(B,X)]).           %  delete list: B's not on X and Y's not clear.
```

9. Partial order planning (15)

Consider using a partial-order planner for the development and release of a software product. Assume there exist the following operators defined in the Strips format:

```
% operator(Operator, Preconditions, Adds, Deletes).
operator(code,[],[haveCode,buggy],[notBuggy]).
operator(optimize,[haveCode],[optimized,buggy],[notBuggy]).
operator(debug,[havecode],[notBuggy],[buggy]).
operator(ship,[havecode,notBuggy,optimized,haveBox],[rich],[]).
operator(designBox,[haveCode],[haveBox],[]).
```

Assume the start state has no initial effects and the finish state has the precondition rich. (a) Show a partial order plan that might be developed by a partial order planner. Be sure to show all of the causal links and temporal links introduced to resolve threats. (10); (b) Show a total ordering of the steps in your partial order plan. (5)

10. Decision trees (30)

You've been hired to help a studio figure out how to create hit television shows. To this end, you plan to analyze the attributes of past shows and induce a decision tree to predict whether or not a proposed new show will be a hit. The C5 schema and data files you develop are:

| tv.names is the schema file.
Hit.
Type: comedy, drama, news, sports.
Location: LA, NYC, other.
Duration: 30, 60.
Hit: yes,no.

| tv.data is training data.
comedy, NYC, 30, yes
comedy, LA, 60, no
news, other, 60, yes
news, LA, 30, yes
sports, other, 60, no
Drama, LA, 60 no

Useful facts about lg, log base 2:
lg(a*b)=lg(a)+lg(b), lg(a/b)=lg(a)-lg(b), lg(0)=-infinity, lg(1/2)=-1, lg(1/3)=-1.58, lg(2/3)=-0.58, lg(1)=0, lg(2)=1, lg(3)=1.58, lg(4)=2, lg(5)=2.32, lg(6)=2.58

(a) What is the **entropy** of the initial set of training data, e.g. the six examples in which three are positive and three negative? (10) Show all of your work. Recall that the entropy of a set of instances S with p positive and n negative cases is defined as:

$$I(p,n) = - (p/(p+n))*lg(p/(p+n)) - (n/(p+n))*lg(n/(p+n))$$

(b) Compute the **information gain** for each of the three attributes (type, location, duration) that could be used for the root of the decision tree. (15) Show all of your work. Recall that the information gain obtained if we ask about attribute A is just the difference between the initial entropy and the weighted sum of the entropy in each of the children of the new node:

$$\text{Gain}(S, A) = I(p,n) - \sum\left((pj+nj)/(p+n)\right) * I(pj,nj)$$

Where the summation is over the children of the node and pj is the number of positive cases in the jth child and nj is the number of negative cases in the jth child.

(c) Show the final **decision tree** that C5 would produce if it did no pruning. (5)