(5/3/02)

Total 53 points. Good luck! You'll need it

(6)
1. Let $s_1 = \{X := g(Y), Y := W\}$ and $s_2 = \{Y := Z, Z := f(X,Y)\}$ be substitutions.
(a) Write $s_1 s_2$.

(b) Write $s_2 s_1$.

(4)
2. Step by step, transform the following wff into an equivalent prenex form.
$(\forall X)p(X) \Rightarrow (\forall Y)q(X,Y)$

(6)
3. Show step by step the following by resolution refutation.
$p \Rightarrow q \wedge r,\ q \Leftrightarrow s \models p \Rightarrow s \wedge r$

(5)

4. Using the method we discussed in class, convert the following wff to a clause set. Show every step.

$\neg(\forall W)\neg(\forall X)\neg(p(X,W)\Rightarrow(\forall Y)q(X,Y,W))$

(1+1+2)

5. For each, write all the outputs from PROLOG.

(a) no program
query: A=3+4, A is 7.

(b) no program
query: A=3, A is A+1.

(c) program: favorite(Ann,cake,pear,Soup).
query: favorite(dave,Pie,Lemmon,X).

(6)

6. Consider the following Prolog program:

```
a(X,Z) :- b(Y,X), c(Z).
b(X,Z) :- d(Z,X).
b(e,f).
c(X) :- d(X,e).
d(g,f).
c(f).
d(f,e).
```

Show the **entire** search tree with bindings to the query 'a(A,f), d(g,A).' Write also the outputs.

(3+4+5+5+5)

7. Write a PROLOG program for each of the following.

(a) Define **only one** predicate which, for a given list, checks if it has an odd number of elements.

(b) Define **only one** predicate to replace zero or more occurrences of given X with given Y (you may assume $Y \neq X$) in a given list.

(c) Define **only one** predicate which, given a list L, an object x, and a positive integer k, creates a copy of L with x inserted at the k-th position. e.g., $[a_1,a_2,a$ x, and k=2 ==> $[a_1,x,a_2,a_3]$. If the length of L is less than k, insert at the end.

(d) To compute the following function $f : \mathbb{N} \rightarrow \mathbb{N}$ **iteratively**.
$f(n) = n+1$ if $n<2$ and $f(n) = 3f(n-1) - f(n-2)$ otherwise

(e) To check if a given string represented as a list is in language $\{b^n c^{3n+2} \mid n \in \mathbb{N}\}$. Use **difference-lists**. No append.