CSCI 162 Lab 8

Prolog Lists

Purpose: To Connect Prolog to Predicate Logic.

Objectives: After this lab you should be able to

- Use lists in Prolog.

About Prolog lists:

You can refer to the head and tail (the head is an element, the tail is a list which is the remainder of the list when the head is removed) by formatting it in your structure. For example,

```
sum([], 0).
sum([H], H).
sum([H|T],X) :- sum(T,Tailsum), X is H+Tailsum.
```

The symbol '|' in the square brackets breaks the list up into the head (an element) and the tail (a list). If the list is empty, when this is done, you'll get `No' -- i.e, it cannot be satisfied.

However, notice that there is a sum clause, above, that deals with the empty list; Prolog will check the clauses in order, so it will check the first clause first, and will only try the others if so far it is unsatisfied.

Now you can refer to H and T in your structure. H and T are just variables. We could have used X and Y.

Note: no typechecking is done in the above. It is a good idea to do typechecking using number(X). For example, sum could be:

```
sum([], 0).
sum([H], H):- number(H).
sum([H|T],X) :- number(H), sum(T,Tailsum), X is H+Tailsum.
```

Instructions:

1. Write a Prolog program that finds the maximum in a list.

It should be called by the following query:

```
max([3, 19, 2, ... 70], X).
```

What to hand in:

Hand in a listing of your file via Moodle.