Computer Engineering 171
Homework 4: Logic Programming

Due: December 4th at 9:00 am

In this assignment, you will write programs in Prolog, a logic programming language. Use either the Linux or OS X machines in the Computing Center for this project. The Prolog interpreter is `gprolog`. Choose any two of the following three programs.

1 Quicksort

In Prolog, write a program to sort a list using the ***quicksort*** sorting algorithm discussed previously. Call this program `sort.pro`. Your program should contain at least the following relations:

- `quicksort(X,Y)`: Succeeds if `Y` can be unified with a list containing the sorted elements of `X`.

Goal: To gain experience with list processing in a logical language.

Hints: Write a relation `partition` that partitions a list into two lists, one containing the elements below the pivot element, and another containing the elements above the pivot element.

2 Binary Search Trees

As in the first assignment, a ***binary search tree*** is either empty, or it consists of a node with two binary search trees as subtrees. Each node holds an integer. In Prolog, write a program to implement binary search trees. Call your program `tree.pro`. Your program should contain at least the following relations:

- `insert(X,Y,Z)`: Succeeds if `Z` can be unified to a tree that is the result of inserting the value `X` into `Y`.
- `exists(X,Y)`: Succeeds if the integer value `X` is present in the tree `Y`.

Goal: To represent structures as terms in Prolog.

Hints: A tree node can be represented using terms such as `node/3` and `empty/0`.

3 The 8-Queens Problem

Write a Prolog program to solve the ***8-queens problem*** from the second assignment. Your program should be capable of backtracking so that all possible solutions can be generated. Call this program `queens.pro`. Your program should contain at least the following relations:

- `queens(X)`: Succeeds if `X` can be unified to a chessboard containing a solution to problem of placing eight queens on the board such that no queen can attack another.

Goal: To learn how backtracking works in Prolog.

Hints: Represent the board as a list of positions. Each position is a term consisting of the row and column number of a queen such as `R/C` or `piece(R,C)`. Since each queen must be placed in a different row, unify each row variable with its row number so that you need only solve for the column number.