Prolog Site

Search this site

Prolog Problems >

7. Miscellaneous

Solutions can be found here.

7.01 (**) Eight queens problem

This is a classical problem in computer science. The objective is to place eight queens on a chessboard so that no two queens are attacking each other; i.e., no two queens are in the same row, the same column, or on the same diagonal.

Hint: Represent the positions of the queens as a list of numbers 1..N. Example: [4,2,7,3,6,8,5,1] means that the queen in the first column is in row 4, the queen in the second column is in row 2, etc. Use the generate-and-test paradigm.

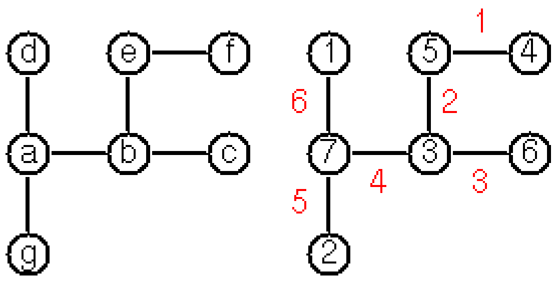
7.02 (**) Knight's tour

Another famous problem is this one: How can a knight jump on an NxN chessboard in such a way that it visits every square exactly once?

Hints: Represent the squares by pairs of their coordinates of the form X/Y, where both X and Y are integers between 1 and N. (Note that '/' is just a convenient functor, not division!) Define the relation jump(N,X/Y,U/V) to express the fact that a knight can jump from X/Y to U/V on a NxN chessboard. And finally, represent the solution of our problem as a list of N*N knight positions (the knight's tour).

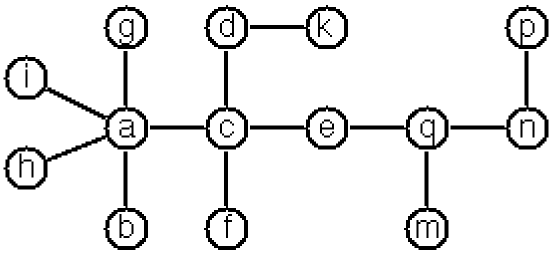
7.03 (***) Von Koch's conjecture

Several years ago I met a mathematician who was intrigued by a problem for which he didn't know a solution. His name was Von Koch, and I don't know whether the problem has been solved since.



Anyway, the puzzle goes like this: Given a tree with N nodes (and hence N-1 edges). Find a way to enumerate the nodes from 1 to N and, accordingly, the edges from 1 to N-1 in such a way, that for each edge K the difference of its node numbers equals to K. The conjecture is that this is always possible.

For small trees the problem is easy to solve by hand. However, for larger trees, and 14 is already very large, it is extremely difficult to find a solution. And remember, we don't know for sure whether there is always a solution!



Write a predicate that calculates a numbering scheme for a given tree. What is the solution for the larger tree pictured above?

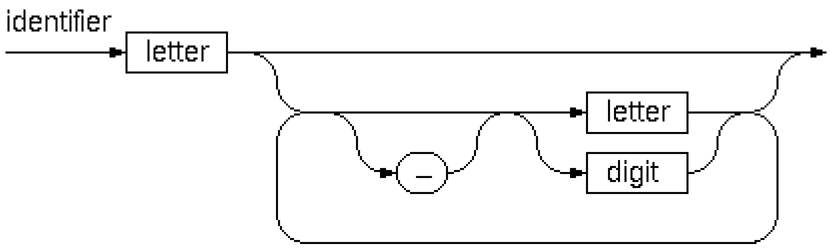
7.04 (***) An arithmetic puzzle

Given a list of integer numbers, find a correct way of inserting arithmetic signs (operators) such that the result is a correct equation. Example: With the list of numbers [2,3,5,7,11] we can form the equations 2-3+5+7 = 11 or 2 = (3*5+7)/11 (and ten others!).

7.05 (**) English number words

On financial documents, like cheques, numbers must sometimes be written in full words. Example: 175 must be written as one-seven-five. Write a predicate full_words/1 to print (non-negative) integer numbers in full words.

7.06 (**) Syntax checker



In a certain programming language (Ada) identifiers are defined by the syntax diagram (railroad chart) opposite. Transform the syntax diagram into a system of syntax diagrams which do not contain loops; i.e. which are purely recursive. Using these modified diagrams, write a predicate identifier/1 that can check whether or not a given string is a legal identifier.

% identifier(Str) :- Str is a legal identifier

7.07 (**) Sudoku

Sudoku puzzles go like this:

```
   Problem statement                   Solution

    .  .  4 | 8  .  . | .  1  7      9  3  4 | 8  2 
```

```
        |         |                    |
6  7  . | 9  .  . | .  .  .      6  7  2 | 9  1
        |         |                    |
5  .  8 | .  3  . | .  .  4      5  1  8 | 6  3
--------+---------+--------      --------+-------
3  .  . | 7  4  . | 1  .  .      3  2  5 | 7  4
        |         |                    |
.  6  9 | .  .  . | 7  8  .      4  6  9 | 1  5
        |         |                    |
.  .  1 | .  6  9 | .  .  5      7  8  1 | 2  6
--------+---------+--------      --------+-------
1  .  . | .  8  . | 3  .  6      1  9  7 | 5  8
        |         |                    |
.  .  . | .  .  6 | .  9  1      8  5  3 | 4  7
        |         |                    |
2  4  . | .  .  1 | 5  .  .      2  4  6 | 3  9
```

Every spot in the puzzle belongs to a (horizontal) row and a (vertical) column, as well as to one single 3x3 square (which we call "square" for short). At the beginning, some of the spots carry a single-digit number between 1 and 9. The problem is to fill the missing spots with digits in such a way that every number between 1 and 9 appears exactly once in each row, in each column, and in each square.

7.08 (***) Nonograms

Around 1994, a certain kind of puzzles was very popular in England. The "Sunday Telegraph" newspaper wrote: "Nonograms are puzzles from Japan and are currently published each week only in The Sunday Telegraph. Simply use your logic and skill to complete the grid and reveal a picture or diagram." As a Prolog programmer, you are in a better situation: you can have your computer do the work!

The puzzle goes like this: Essentially, each row and column of a rectangular bitmap is annotated with the respective lengths of its distinct strings of occupied cells. The person who solves the puzzle must complete the bitmap given only these lengths.

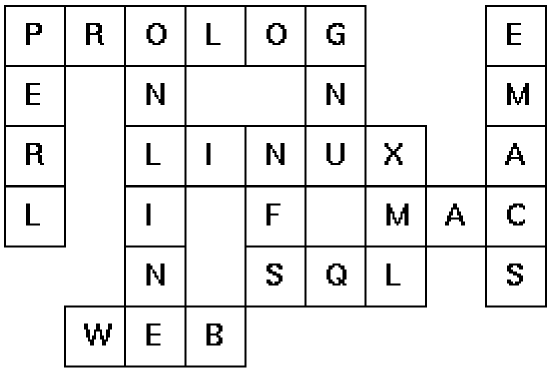
```
Problem statement:          Solution:

|_|_|_|_|_|_|_|_| 3         |_|X|X|X|_|_|_|_
|_|_|_|_|_|_|_|_| 2 1       |X|X|_|X|_|_|_|_
|_|_|_|_|_|_|_|_| 3 2       |_|X|X|X|_|_|X|X
|_|_|_|_|_|_|_|_| 2 2       |_|_|X|X|_|_|X|X
|_|_|_|_|_|_|_|_| 6         |_|_|X|X|X|X|X|X
|_|_|_|_|_|_|_|_| 1 5       |X|_|X|X|X|X|X|_
|_|_|_|_|_|_|_|_| 6         |X|X|X|X|X|X|_|_
|_|_|_|_|_|_|_|_| 1         |_|_|_|_|X|_|_|_
|_|_|_|_|_|_|_|_| 2         |_|_|_|X|X|_|_|_
 1 3 1 7 5 3 4 3             1 3 1 7 5 3 4
 2 1 5 1                     2 1 5 1
```

For the example above, the problem can be stated as the two lists [[3],[2,1],[3,2],[2,2],[6],[1,5],[6],[1],[2]] and [[1,2],[3,1],[1,5],[7,1],[5],[3],[4],[3]] which give the "solid" lengths of the rows and columns, top-to-bottom and left-to-right, respectively. Published puzzles are larger than this example, e.g. 25 x 20, and apparently always have unique solutions.

7.09 (***) Crossword puzzle

Given an empty (or almost empty) framework of a crossword puzzle and a set of words. The problem is to place the words into the framework.



The particular crossword puzzle is specified in a text file which first lists the words (one word per line) in an arbitrary order. Then, after an empty line, the crossword framework is defined. In this framework specification, an empty character location is represented by a dot (.). In order to make the solution easier, character locations can also contain predefined character values. The puzzle opposite is defined in the file p7_09a.dat, other examples are p7_09b.dat and p7_09d.dat. There is also an example of a puzzle (p7_09c.dat) which does not have a solution.

Words are strings (character lists) of at least two characters. A horizontal or vertical sequence of character places in the crossword puzzle framework is called a *site*. Our problem is to find a compatible way of placing words onto sites.

Hints:
1) The problem is not easy. You will need some time to thoroughly understand it. So, don't give up too early! And remember that the objective is a clean solution, not just a quick-and-dirty hack!
(2) Reading the data file is a tricky problem for which a solution is provided in the file p7_09-readfile.pl. Use the predicate read_lines/2.
(3) For efficiency reasons it is important, at least for larger puzzles, to sort the words and the sites in a particular order. For this part of the problem, the solution of 1.28 may be very helpful.

Subpages (1): Solutions-7