CS-125 Logic Programming

Exercises

**Question 1.** Define a predicate `last/2` such that `last(X,L)` means that `X` is the last member of the list `L`.

**Question 2.** Define a predicate `numeral/1` that tests whether an object is a numeral, that is, one of the terms `0, s(0), s(s(0)), ...`

**Question 3.** Write a program for multiplication of two natural numbers, where the natural numbers are represented by numerals `0, s(0), s(s(0)), ...`. The program should include the program `SUM`, and it should define a predicate `prod/3` such that `prod(X,Y,Z)` means `X*Y=Z`. To find the defining clauses for `prod` consider the following recursive definition of multiplication:

\[
\begin{align*}
n * 0 &= 0 \\
n * (m + 1) &= n * m + n
\end{align*}
\]

**Question 4.** Define a predicate `len/2` that computes the length of a list as a numeral.

**Question 5.** Define a predicate `middle/2` that computes the ‘middle element’ of a list with an odd number of elements. If the list has an even number of elements the predicate should fail.

**Question 6.** Find a list of 27 numbers such that each of the numbers `1,\ldots,9` occurs three times and between the occurrences of a number `n` there are `n` other numbers. Write a Prolog program that solves this puzzle.

Hint: Describe the solution `L` as follows: `L` has length 27 and contains sublists of the form `[1,_,1,_,1]`, `[2,_,_,_,2,_,_,_,2]`, and so on.

The predicates should be tested by appropriate queries.

At least one of the exercises needs to be solved in a lab session.