PFL - HW labs

Ex 4

a) Determine the positions of the maximal element of a linear list

flow (i, i, o)

- maxNumbers (A, B, A) :- A >= B.
  maxNumbers (A, B, B) :- A < B.

$$maxNumbers(A, B) = \begin{cases} A; & A >= B \\ B; & A < B \end{cases}$$

flow (i, o)

- maxList ([H], H) :- !

maxList ([H|T], R) :-
    maxList (T, R1),
    maxNumbers (H, R1, R).

$$maxList(l_1, \dots, l_n) = \begin{cases} 0; & n = 0 \\ l_1; & n = 1 \\ maxNumbers(l_1, maxList(l_2, \dots, l_n)) & \text{otherwise} \end{cases}$$

flow(i,i,o,i)

```
- allPos([],_,[],_).
allPos([E|T], E, [i|is], i):-
    i1 is i+1,
    allPos(T, E, is, i1).
allPos([H|T], E, is, i):- dif(H,E),
    i1 is i+1,
    allPos(T, E, is, i1).
```

$$allPos((l_1 \ldots l_n), E, i) = \begin{cases} \emptyset & n=0 \\ ~~allPos((l_2 \ldots l_n), E, R, i)~~ \\ i \cup allPos(l_2 \ldots l_n, E, i_1), & l_1 = E \\ allPos((l_2 \ldots l_n), E, i_1), & l_1 \neq E \end{cases}$$

(i=0; i_1 → i++)

~~findAllMaxPos = allPos(L, maxList~~

~~findAllMaxPos(L) = allPos(L, maxList(L))~~

~~findAllMaxPos($l_1 \ldots l_n$) = allPos~~

flow(i,o)

```
- findAllMaxPos(L, is):-
    maxList(L, R),
    allPos(L, R, is, 0)
```

b) for a heterogenous list formed from integers and lists of integers, replace all sublists with the list made out of the positions of the indexes of the max element from that sublist

flow(i,o)

listProcess([],[]).

listProcess([H|T],[H|R]):-number(H),
listProcess(T,R).

listProcess([H|T],[L1|R]):isList(H),
findAllMaxPos(H,L1),
listProcess(T,R).

$$listProcess(l_1 \dots l_n) = \begin{cases} \emptyset; n=0 \\ l_1 \cup listProcess(l_2 \dots l_n); & l_1 \text{ is number} \\ findAllMaxPos(l_1) \cup listProcess(l_2 \dots l_n); & l_1 \text{ is list} \end{cases}$$

→ findAllMaxPos(L) = (allPos(L, maxList(L))