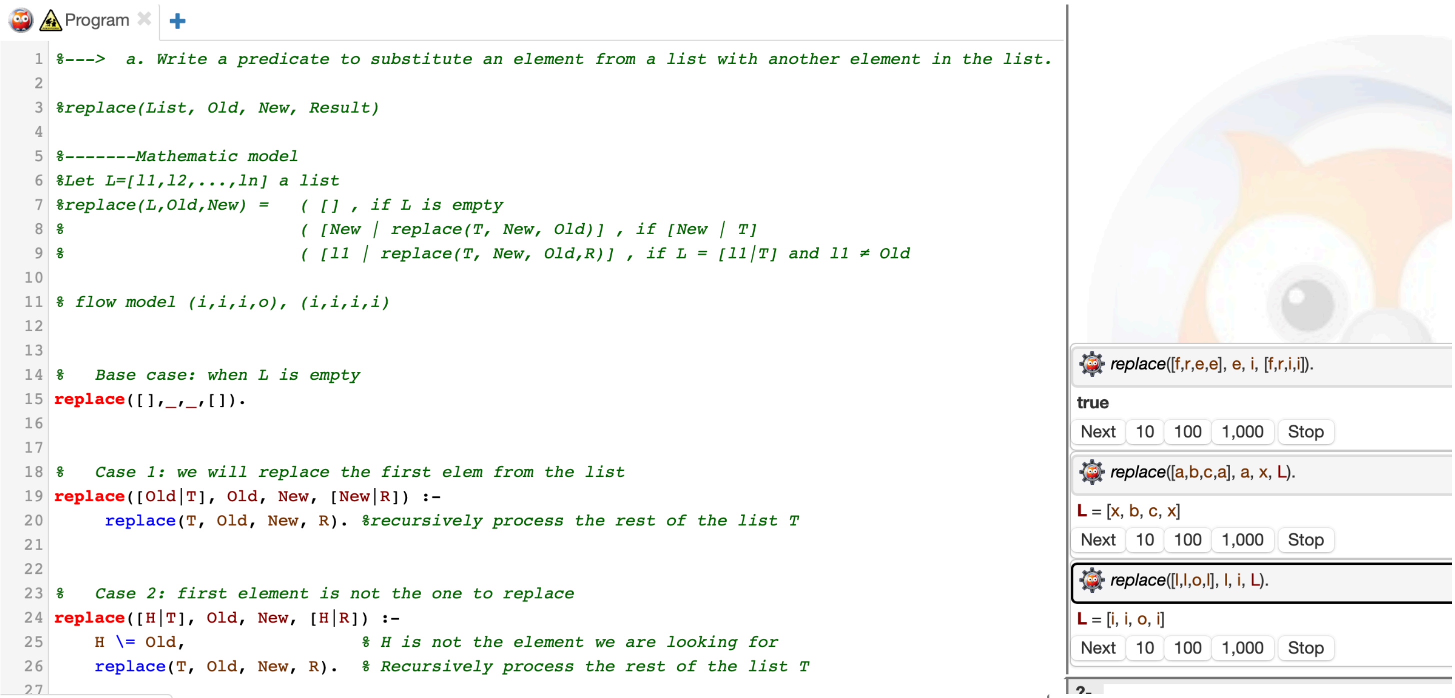
LAB2 ———> Prolog1



```
%--->  a. Write a predicate to substitute an element from a list with another element in the list.

%replace(List, Old, New, Result)

%-------Mathematic model
%Let L=[l1,l2,...,ln] a list
%replace(L,Old,New) =    ( [] , if L is empty
%                        ( [New | replace(T, New, Old)] , if [New | T]
%                        ( [l1 | replace(T, New, Old,R)] , if L = [l1|T] and l1 ≠ Old

% flow model (i,i,i,o), (i,i,i,i)


%   Base case: when L is empty
replace([],_,_,[]).


%   Case 1: we will replace the first elem from the list
replace([Old|T], Old, New, [New|R]) :-
     replace(T, Old, New, R). %recursively process the rest of the list T


%   Case 2: first element is not the one to replace
replace([H|T], Old, New, [H|R]) :-
    H \= Old,                  % H is not the element we are looking for
    replace(T, Old, New, R).   % Recursively process the rest of the list T
```

replace([f,r,e,e], e, i, [f,r,i,i]).

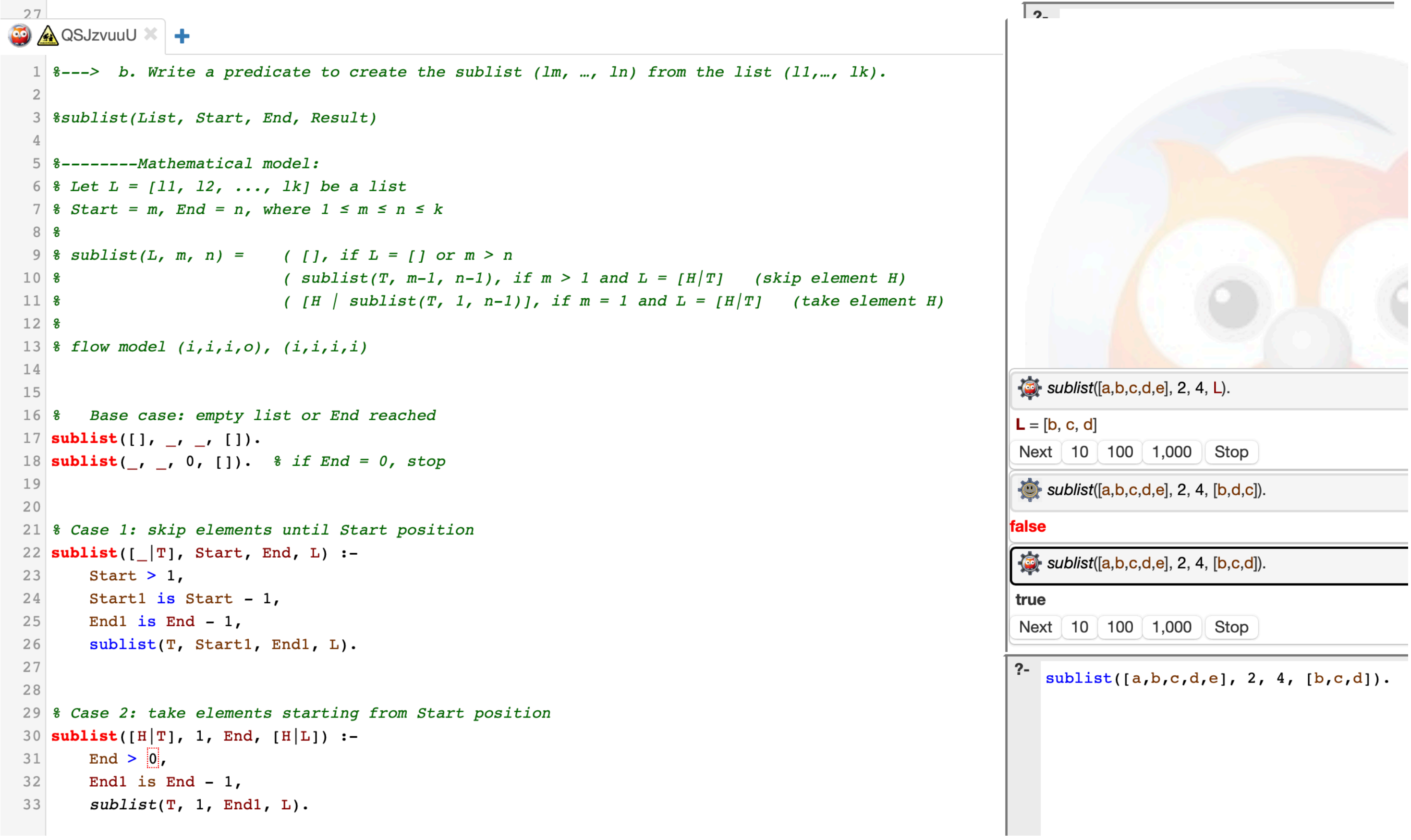
true

replace([a,b,c,a], a, x, L).

L = [x, b, c, x]

replace([l,l,o,l], l, i, L).

L = [i, i, o, i]



```
%--->  b. Write a predicate to create the sublist (lm, …, ln) from the list (l1,…, lk).

%sublist(List, Start, End, Result)

%--------Mathematical model:
% Let L = [l1, l2, ..., lk] be a list
% Start = m, End = n, where 1 ≤ m ≤ n ≤ k
%
% sublist(L, m, n) =    ( [], if L = [] or m > n
%                       ( sublist(T, m-1, n-1), if m > 1 and L = [H|T]   (skip element H)
%                       ( [H | sublist(T, 1, n-1)], if m = 1 and L = [H|T]   (take element H)
%
% flow model (i,i,i,o), (i,i,i,i)


%   Base case: empty list or End reached
sublist([], _, _, []).
sublist(_, _, 0, []).  % if End = 0, stop


% Case 1: skip elements until Start position
sublist([_|T], Start, End, L) :-
    Start > 1,
    Start1 is Start - 1,
    End1 is End - 1,
    sublist(T, Start1, End1, L).


% Case 2: take elements starting from Start position
sublist([H|T], 1, End, [H|L]) :-
    End > 0,
    End1 is End - 1,
    sublist(T, 1, End1, L).
```

sublist([a,b,c,d,e], 2, 4, L).

L = [b, c, d]

sublist([a,b,c,d,e], 2, 4, [b,d,c]).

false

sublist([a,b,c,d,e], 2, 4, [b,c,d]).

true

```
?- sublist([a,b,c,d,e], 2, 4, [b,c,d]).
```