

Prolog : TP6

jeudi 6 novembre 2014

Maud LERAY
Corentin NICOLE
groupe 1.2

Questions

```
%% TP6
%% Damien Crmilleux, Lauriane Holy
%%%%%%%%%%%%%% First part
copy_prog(program(
start,
[stop],
[delta(start, ' ', ' ', right, stop),
delta(start, 1, ' ', right, s2),
delta(s2, 1, 1, right, s2),
delta(s2, ' ', ' ', right, s3),
delta(s3, 1, 1, right, s3),
delta(s3, ' ', 1, left, s4),
delta(s4, 1, 1, left, s4),
delta(s4, ' ', ' ', left, s5),
delta(s5, 1, 1, left, s5),
delta(s5, ' ', 1, right, start)
]
)
).

%next(+, +, +, -, -, -)
next(Program, StatePrev, SymbolPrev, SymbolNext, Dir, StateNext) :-
    transitions(Program, Deltas),
    nextFromDeltas(Deltas, StatePrev, SymbolPrev, SymbolNext, Dir, StateNext).

nextFromDeltas([delta(StatePrev, SymbolPrev, SymbolNext, Dir, StateNext) | _],
    StatePrev, SymbolPrev, SymbolNext, Dir, StateNext).

nextFromDeltas([_|Rdeltas], StatePrev, SymbolPrev, SymbolNext, Dir, StateNext) :-
    nextFromDeltas(Rdeltas, StatePrev, SymbolPrev, SymbolNext, Dir, StateNext).

% Update_tape(+, +, +, -)
update_tape(tape([E], [_|R]), Symbol, left, tape([' '], [E | [Symbol | R]])) :-
    !.

update_tape(tape(LL, [_]), Symbol, right, tape(RLL, [' '])) :-
    addLast(LL, Symbol, RLL),
    !.

update_tape(tape(LL, [_|R]), Symbol, right, tape(RLL, R)) :-
    addLast(LL, Symbol, RLL).

update_tape(tape(LL, [_|ResteLR]), Symbol, left, tape(NLL, [E | [Symbol | ResteLR]])) :-
    removeLast(LL, NLL, E).

addLast([], E, [E]).
addLast([T|Q], E, [T|R]) :-
    addLast(Q, E, R).

removeLast([E], [], E).
removeLast([T|Q], [T|R], E) :-
```

```
        removeLast(Q,R,E) .

% run_turing_machine(+, +, -, -): program * symbol list * symbol list * state
% run_turing_machine(Program, Input, Output, FinalState)
conc2([],L,L) .
conc2([Tete|Reste],L2,[Tete|R]) :-conc2(Reste,L2,R) .

teteDeLecture(tape( _, [Symb|_]), Symb) .

isFinalState(program( _ , [State|_], _), State) :-
    !.
isFinalState(program( _ , [_|Reste], _), State) :-
    isFinalState(program( _ , Reste, _), State) .

run_turing_machine(Program, tape(A, B), Output, FinalState, FinalState) :-
    isFinalState(Program, FinalState),
    conc2(A, B, Output) .

run_turing_machine(Program, Input, Output, FinalState, CurrentState) :-
    teteDeLecture(Input, TLect),
    next(Program, CurrentState, TLect, NSymb, Dir, Nstate),
    update_tape(Input, NSymb, Dir, Ntape),
    run_turing_machine(Program, Ntape, Output, FinalState, Nstate) .

run_turing_machine(Program, Input, Output, FinalState) :-
    initial_state(Program, InitS),
    teteDeLecture(tape([' '], Input), TLect),
    next(Program, InitS, TLect, NSymb, Dir, Nstate),
    update_tape(tape([' '], Input), NSymb, Dir, Ntape),
    run_turing_machine(Program, Ntape, Output, FinalState, Nstate) .

% run_turing_machine2
/*
 * Nous n'avons pas reussie a compiler le programme fournis.
 * erreur de syntaxe d'apres le compilateur GNU prolog.
 */

%%%%%%%%% PROGRAMME DE TEST %%%%%%%%%%
test(0 ,SymbolNext,Dir,StateNext) :-
    copy_prog(Prog),
    next(Prog, s4, 1, SymbolNext,Dir,StateNext) .

test(1, Output, FinalState) :-
    copy_prog(Prog),
    run_turing_machine(Prog, [1], Output, FinalState) .

%%%%%%%%%%%%%%%%%%%%%%%%% TEST %%%%%%%%%%%%%%%%%%%%%%%%%%
/*
[eclipse 42]: test(0,SymbolNext,Dir,StateNext) .

SymbolNext = 1
Dir = left
```

```
StateNext = s4
*/

/*
| ?- update_tape(tape([1,2,3],[' ']),10,right,T).

T = tape([1,2,3,10],[' '])

yes
| ?- update_tape(tape([' '],[3,4]),10,left,T).

T = tape([' '],[' ',10,4])

yes
| ?- update_tape(tape([1,2],[3,4]),10,left,T).

T = tape([1],[2,10,4]) ?

yes
| ?- update_tape(tape([1,2],[3,4]),10,right,E).

E = tape([1,2,10],[4]) ?

yes
| ?- test(1, Output,FinalState).

FinalState = stop
Output = [' ',1,' ',1] ?

*/
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Listing 1: tp6.pl