```prolog
pgm(X) --> cmd(X).
pgm(seq(X1,X2)) --> cmd(X1),[;],pgm(X2).
num(num(X)) --> [num(X)],{integer(X)}.
id(id(X))  --> [id(X)],{atomic(X),\+integer(X)}.

cmd(skip) --> [skip].
cmd(set(X,Y)) --> id(X),[:=],pwer(Y).
cmd(if(B,P1,P2)) --> [if],bool(B),[then],
        pgm(P1),[else],pgm(P2),[fi].
cmd(while(B,P1)) --> [while],bool(B),[do],pgm(P1),[od].

bool(true) --> [true].
bool(false) --> [false].
bool(E1 == E2) --> expr(E1),[=],expr(E2).
bool(E1 > E2) --> expr(E1),[>],expr(E2).
bool(E1 < E2) --> expr(E1),[<],expr(E2).

pwer(-X) --> [-],pwer(X).
pwer(X) --> expr(X).
expr(E1 * E2) --> factor(E1),[*],expr(E2).
expr(X) --> factor(X).
factor(E1 + E2) --> term(E1),[+],factor(E2).
factor(E1 - E2) --> term(E1),[-],factor(E2).
factor(X) --> term(X).
term(X) --> num(X).
term(X) --> id(X).

parse(Tokens, SyntaxTree) :-
    phrase(pgm(SyntaxTree), Tokens).
```