```
% parser.pl
% expr(Z,"-2+3*5+1",[]) -> Z=14
expr(Z) --> term(X), "+", expr(Y), {Z is X + Y}.
expr(Z) --> term(X), "-", expr(Y), {Z is X - Y}.
expr(X) --> term(X).

term(Z) --> number(X), "*", term(Y), {Z is X * Y}.
term(Z) --> number(X), "/", term(Y), {Z is X / Y}.
term(Z) --> number(Z).

number(C) --> "+", number(C).
number(C) --> "-", number(X),{C is -X}.
number(X) --> [C], {"0" =< C, C =< "9", X is C - "0"}.

/*

expr(A, B, C) :-
        term(F, B, G),
        'C'(G, 43, E),
        expr(D, E, C),
        A is F+D.
expr(A, B, C) :-
        term(F, B, G),
        'C'(G, 45, E),
        expr(D, E, C),
        A is F-D.
expr(A, B, C) :-
        term(A, B, C).

number(A, B, C) :-
        'C'(B, 43, D),
        number(A, D, C).
number(A, B, C) :-
        'C'(B, 45, E),
        number(D, E, C),
        A is-(D).
number(A, B, C) :-
        'C'(B, D, C),
        [48]=<D,
        D=<[57],
        A is D-[48].

library_directory('/opt/net1/sicstus3/bin/library').

file_search_path(library, A) :-
        library_directory(A).

term(A, B, C) :-
        number(F, B, G),
        'C'(G, 42, E),
        term(D, E, C),
        A is F*D.
term(A, B, C) :-
        number(F, B, G),
        'C'(G, 47, E),
        term(D, E, C),
        A is F/D.
term(A, B, C) :-
        number(A, B, C).
*/
```

```
expr --> term, "+", expr.
expr --> term, "-", expr.
expr --> term.

term --> number, "*", term.
term --> number, "/", term.
term --> number.

number --> "+", number.
number --> "-", number.
number --> [C], {"0" =< C, C =< "9"}.

/*
| ?-  expr("-2+3*5+1",[]).

yes
| ?- expr("-2+3*5+11",[]).

no
| ?-  expr("-2+3*5+11","+11").


*/
```

```
expr --> term, "+", expr.
expr --> term, "-", expr.
expr --> term.

term --> number, "*", term.
term --> number, "/", term.
term --> number.

number --> "+", digits.
number --> "-", digits.
number -->  digits.

digits  --> digit.
digits  --> [].
digits  --> digit, digits.

digit --> [C], {"0" =< C, C =< "9"}.

/*
| ?-  expr("-2+3*5+1",[]).

yes
?- expr("-245+3*5+11",[]).

yes

*/
```

```
expr(Z) --> term(X), "+", expr(Y), {Z is X + Y}.
expr(Z) --> term(X), "-", expr(Y), {Z is X - Y}.
expr(X) --> term(X).

term(Z) --> number(X), "*", term(Y), {Z is X * Y}.
term(Z) --> number(X), "/", term(Y), {Z is X / Y}.
term(Z) --> number(Z).

number(C) --> "+", number(C).
number(C) --> "-", number(X),{C is -X}.
number(X) --> [C], {"0" =< C, C =< "9", X is C - "0"}.

/*

| ?- expr(Z,"-2+3*5+1",[]).

Z = 14 ? ;

no
| ?- expr(Z,"-2+3*5+11",[]).

no
| ?- expr(Z,"-2+3*5+11","+11").

Z = 13 ?

yes


*/
```

```
expr(Z) --> term(X), "+", expr(Y), {Z is X + Y}.
expr(Z) --> term(X), "-", expr(Y), {Z is X - Y}.
expr(X) --> term(X).

term(Z) --> number(X), "*", term(Y), {Z is X * Y}.
term(Z) --> number(X), "/", term(Y), {Z is X / Y}.
term(Z) --> number(Z).

number(C) --> "+", digits(C,_).
number(C) --> "-", digits(X,_),{C is -X}.
number(C) -->  digits(C,_).

digits(C,1) --> digit(C).
digits(V,B)  --> digit(X), digits(C,B0),{B is 10*B0, V is X*B +C}.
digit(X) --> [C], {"0" =< C, C =< "9", X is C - "0"}.

/*

?-  expr(Z,"-2+3*5+11",[]).

Z = 24 ? ;

no
| ?-  expr(Z,"-222+3*5+11",[]).

Z = -196 ? ;

no

| ?- listing.

digit(A, B, C) :-
        'C'(B, D, C),
        [48]=<D,
        D=<[57],
        A is D-[48].

digits(A, 1, B, C) :-
        digit(A, B, C).
digits(A, B, C, D) :-
        digit(E, C, F),
        digits(G, H, F, D),
        B is 10*H,
        A is E*B+G.
...
*/
```