PARSER EXAMPLE 3

MAKEFILE

```
flex scanner.l
bison   -d  parser.y
gcc -o parser lex.yy.c  parser.tab.c -lfl.
```

SCANNER.L

```
%{
    # include "parser.tab.h"
    cnt lineno = 0;
%}


%%

[ \t]                { /* eat white spaces */ }
[a-z A-Z][a-z A-Z]*     { return ID; }
[0-9][0-9]*             { return NUMBER; }
[+-]                    { return yytext[0]; }
\n                      { ++lineno; return '\n'; }
.                       { printf ("line no: %d : unrecognized character
                          '%c' \n", lineno, yytext[0]); }
```

☆ PARSER·Y

```
%{
  #include <stdio.h>
%}

%token  NUMBER  ID

%%

Goal: Expr '\n';

Expr: Expr '+' Term
    | Expr '-' Term
    | Term ;

Term: NUMBER
    | ID;

%%

yyerror (char *s) {
  fprintf (stderr, "error: %s \n", s);
}

main (int argc, char ** argv) {
  yyparse ();
  printf ("parsing over \n");
}
```