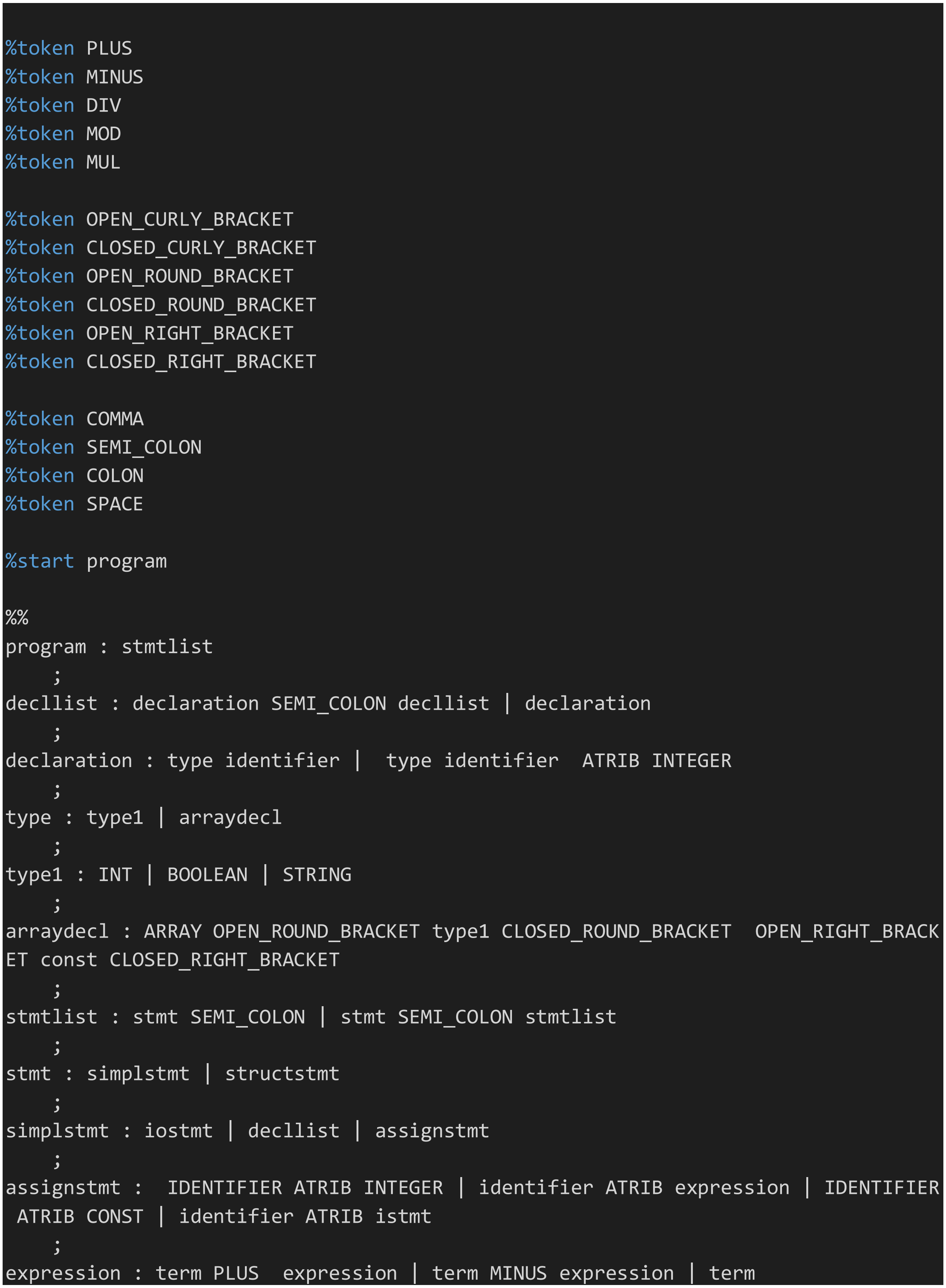
Tiutiu Natan-Gabriel

https://github.com/Natan-Gabriel/FLCD

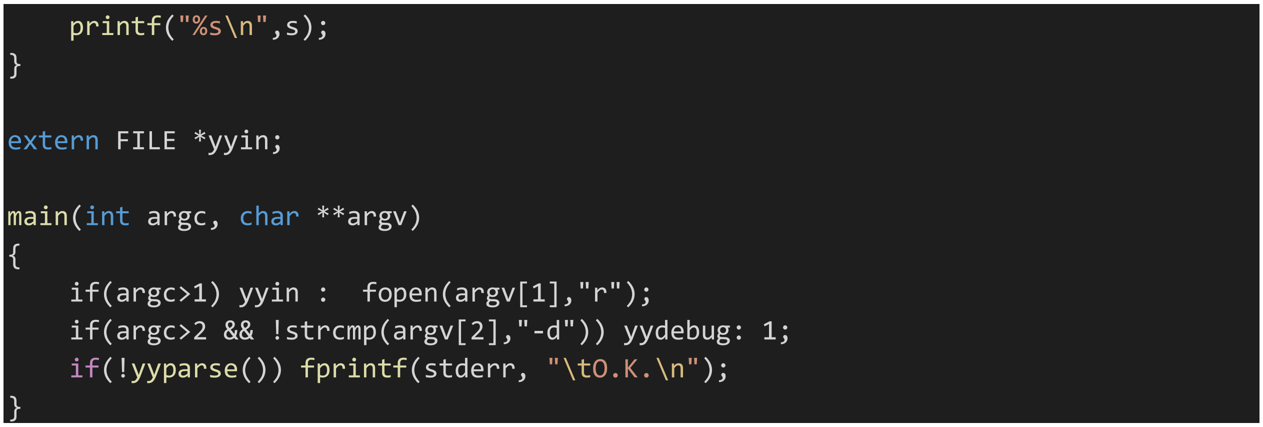
Spec.y file:

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

#define YYDEBUG 1
%}

%token AND
%token ARRAY
%token FOR
%token IF
%token ELIF
%token ELSE
%token OR
%token PRINT
%token READ
%token READINTEGER
%token INT
%token INTEGER
%token BOOLEAN
%token STRING
%token WHILE

%token IDENTIFIER
%token CONST

%token ATRIB
%token EQ
%token NE
%token LE
%token GE
%token LT
%token GT
%token NOT


%left '+' '-'
```



```
%token PLUS
%token MINUS
%token DIV
%token MOD
%token MUL

%token OPEN_CURLY_BRACKET
%token CLOSED_CURLY_BRACKET
%token OPEN_ROUND_BRACKET
%token CLOSED_ROUND_BRACKET
%token OPEN_RIGHT_BRACKET
%token CLOSED_RIGHT_BRACKET

%token COMMA
%token SEMI_COLON
%token COLON
%token SPACE

%start program

%%
program : stmtlist
    ;
decllist : declaration SEMI_COLON decllist | declaration
    ;
declaration : type identifier |  type identifier  ATRIB INTEGER
    ;
type : type1 | arraydecl
    ;
type1 : INT | BOOLEAN | STRING
    ;
arraydecl : ARRAY OPEN_ROUND_BRACKET type1 CLOSED_ROUND_BRACKET  OPEN_RIGHT_BRACKET const CLOSED_RIGHT_BRACKET
    ;
stmtlist : stmt SEMI_COLON | stmt SEMI_COLON stmtlist
    ;
stmt : simplstmt | structstmt
    ;
simplstmt : iostmt | decllist | assignstmt
    ;
assignstmt :  IDENTIFIER ATRIB INTEGER | identifier ATRIB expression | IDENTIFIER ATRIB CONST | identifier ATRIB istmt
    ;
expression : term PLUS  expression | term MINUS expression | term
```

```
    ;
term : factor MUL term | factor DIV  term | factor MOD  term | factor
    ;
factor : OPEN_ROUND_BRACKET expression CLOSED_ROUND_BRACKET | identifier | const
    ;
iostmt : PRINT OPEN_ROUND_BRACKET STRING CLOSED_ROUND_BRACKET  | PRINT OPEN_ROUND_BRACKET const CLOSED_ROUND_BRACKET | PRINT OPEN_ROUND_BRACKET identifier CLOSED_ROUND_BRACKET | istmt
    ;
istmt : READ  OPEN_ROUND_BRACKET CLOSED_ROUND_BRACKET | READINTEGER OPEN_ROUND_BRACKET CLOSED_ROUND_BRACKET SEMI_COLON
    ;
structstmt : ifstmt | whilestmt
    ;
ifstmt : IF OPEN_ROUND_BRACKET condition CLOSED_ROUND_BRACKET OPEN_CURLY_BRACKET stmtlist CLOSED_CURLY_BRACKET  | IF OPEN_ROUND_BRACKET condition CLOSED_ROUND_BRACKET OPEN_CURLY_BRACKET stmtlist CLOSED_CURLY_BRACKET  elseIfBranches
    ;
elseIfBranches : ELIF OPEN_ROUND_BRACKET condition OPEN_ROUND_BRACKET OPEN_CURLY_BRACKET stmtlist CLOSED_CURLY_BRACKET  | ELIF OPEN_ROUND_BRACKET condition CLOSED_ROUND_BRACKET  OPEN_CURLY_BRACKET stmtlist CLOSED_CURLY_BRACKET  elseIfBranches | elseBranch
    ;
elseBranch : ELSE OPEN_CURLY_BRACKET stmtlist CLOSED_CURLY_BRACKET
    ;
whilestmt : WHILE OPEN_ROUND_BRACKET condition CLOSED_ROUND_BRACKET OPEN_CURLY_BRACKET stmtlist CLOSED_CURLY_BRACKET
    ;
simplecondition : expression relation expression
    ;
condition : simplecondition LogicOPERATOR condition | simplecondition
    ;
LogicOPERATOR : AND  | OR
    ;
relation : LT | LE | ATRIB | EQ | NE | GE | GT
    ;
identifier : IDENTIFIER
    ;
const : CONST
    ;


%%
yyerror(char *s)
{
```



```
    printf("%s\n",s);
}

extern FILE *yyin;

main(int argc, char **argv)
{
    if(argc>1) yyin :  fopen(argv[1],"r");
    if(argc>2 && !strcmp(argv[2],"-d")) yydebug: 1;
    if(!yyparse()) fprintf(stderr, "\tO.K.\n");
}
```