Complier Construction Lab 8

Name: Naivedya Rai	Roll No./PRN: 20070122083	Date: 20/07/23
Class: CS	Division: CS-B	Batch:B1

Desk calculator with error recovery

Code:

Lex:

```
%{
#include <stdio.h>
#include "as8.tab.h"
extern int yylval;
%}

%%
[0-9]+ {
    yylval = atoi(yytext);
    return NUMBER;
}

[\t] ;
[\n] return 0;
. return yytext[0];
%%

int yywrap() {
    return 1;
}
```

Yacc:

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

%token ID NUM FOR LE GE EQ NE OR AND
%right "="
%left OR AND
%left '>' '<' LE GE EQ NE
%left '+' '-'
%left '*' '/'
%right UMINUS
%left '!'
%%

S : ST { printf("Input accepted\n"); exit(0); }

ST : FOR '(' E ';' E2 ';' E ')' DEF
;

DEF : '{' BODY '}'
   | E ';'
   | ST
   |
;

BODY : BODY BODY
   | E ';'
   | ST
   |
;

E : ID '=' E
   | E '+' E
   | E '-' E
   | E '*' E
   | E '/' E
   | E '<' E
   | E '>' E
   | E LE E
   | E GE E
   | E EQ E
   | E NE E
   | E OR E
   | E AND E
   | E '+' '+'
   | E '-' '-'
   | ID
   | NUM
;
```

```
E2 : E'<'E
    | E'>'E
    | E LE E
    | E GE E
    | E EQ E
    | E NE E
    | E OR E
    | E AND E
;
%%

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

int main() {
    printf("Enter the expression:\n");
    yyparse();
}
```

Output:

```
Enter Any Arithmetic Expression which can have operations Addition, Subtraction, Multiplication, Division, Modulus, and
Round brackets:
((2+2)*(4/2)-1)

Result=7

Entered arithmetic expression is Valid
```