2CS701- Compiler Construction

Practical 5

Aim: To implement a calculator in YACC: Syntax Directed Translation.

Use YACC to Write a Grammar for multiple expression statements, and apply syntax directed translation for calculators.

Author: Darshil Maru 20BCE514

Date: September 16, 2022

Guide: Prof. Deepti Saraswat

Calc.l File:

```
%{/*Declarations */
#include <stdlib.h>
void yyerror(char *);
#include "y.tab.h"
%}
%%
[0-9]+      { yylval = atoi(yytext);
              return INTEGER;
            }
[-+*/\n]    return *yytext;
PLUS         return '+';
MINUS        return '-';
MULTIPLY     return '*';
TIMES        return '*';
DIVIDE       return '/';
[ \t] ;
.           yyerror("invalid character");
%%
int yywrap(void) {
return 1;
}
```

Calc.y File:

```
%{/*Declarations */
#include <stdio.h>
int yylex(void);
void yyerror(char *);
%}
%token INTEGER
%%
program:
        program expr '\n' { printf("%d\n", $2); }
        |
        ;
expr:
```

```
        INTEGER { $$ = $1; }
        | expr '+' expr { $$ = $1 + $3; }
        | expr '-' expr { $$ = $1 - $3; }
        | expr '*' expr { $$ = $1 * $3; }
        | expr '/' expr { $$ = $1 / $3; }
        ;
%%
void yyerror(char *s) {
fprintf(stderr, "%s\n", s);
}
int main(void) {
yyparse();
return 0;
}
```

Output: