Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA   Batch - 71
CD Practical 10

<u>Aim</u> : **Implement YACC Program to evaluate a given arithmetic expression.**

Code (p10.l) :

```
%{
    /* Definition section*/
    #include "p10.tab.h"
    extern int yylval;
%}

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

[a-zA-Z]+ { return ID; }
[ \t]+      ; /*For skipping whitespaces*/

\n       { return 0; }
.        { return yytext[0]; }

%%
int yywrap()
{
return 0;
}
```

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA   Batch - 71
CD Practical 10

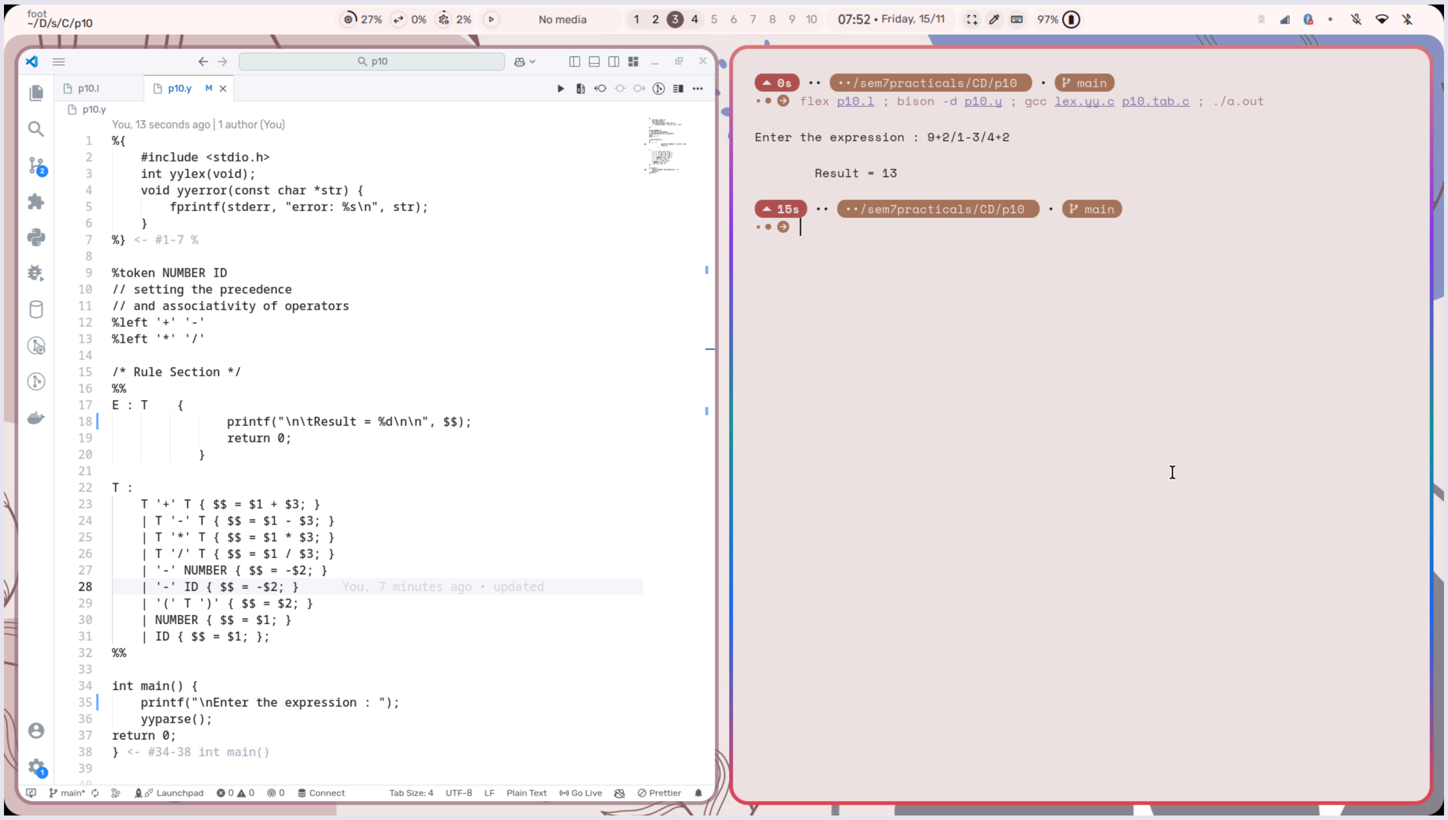
Code (p10.y) :

```
%{
    #include <stdio.h>
    int yylex(void);
    void yyerror(const char *str) {
    fprintf(stderr, "error: %s\n", str);
    }
%}

%token NUMBER ID
// setting the precedence
// and associativity of operators
%left '+' '-'
%left '*' '/'

/* Rule Section */
%%
E : T    {
                printf("\n\tResult = %d\n\n", $$);
                return 0;
            }

T :
    T '+' T { $$ = $1 + $3; }
    | T '-' T { $$ = $1 - $3; }
    | T '*' T { $$ = $1 * $3; }
    | T '/' T { $$ = $1 / $3; }
    | '-' NUMBER { $$ = -$2; }
```

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA   Batch - 71
CD Practical 10

```
    | '-' ID { $$ = -$2; }
    | '(' T ')' { $$ = $2; }
    | NUMBER { $$ = $1; }
    | ID { $$ = $1; };
%%

int main() {
    printf("\nEnter the expression : ");
    yyparse();
return 0;
}
```

Output :



```
%{
    #include <stdio.h>
    int yylex(void);
    void yyerror(const char *str) {
        fprintf(stderr, "error: %s\n", str);
    }
%}

%token NUMBER ID
// setting the precedence
// and associativity of operators
%left '+' '-'
%left '*' '/'

/* Rule Section */
%%
E : T    {
                printf("\n\tResult = %d\n\n", $$);
                return 0;
            }

T :
    T '+' T { $$ = $1 + $3; }
    | T '-' T { $$ = $1 - $3; }
    | T '*' T { $$ = $1 * $3; }
    | T '/' T { $$ = $1 / $3; }
    | '-' NUMBER { $$ = -$2; }
    | '-' ID { $$ = -$2; }
    | '(' T ')' { $$ = $2; }
    | NUMBER { $$ = $1; }
    | ID { $$ = $1; };
%%

int main() {
    printf("\nEnter the expression : ");
    yyparse();
return 0;
}
```

```
▲ 0s •• ••/sem7practicals/CD/p10 • main
•• flex p10.l ; bison -d p10.y ; gcc lex.yy.c p10.tab.c ; ./a.out

Enter the expression : 9+2/1-3/4+2

        Result = 13

▲ 15s •• ••/sem7practicals/CD/p10 • main
••
```