EXPERIMENT 14

Name: - Vishesh Gupta

Roll No.: - 2K18/CO/390

Aim: - Write a program to implement one pass compiler

Code: -

Lexical Analysis Code :-

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

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

     }
[\t] ;

[\n] return 0;

. return yytext[0];

%%

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

Parser Code :-

```
%{
 #include<stdio.h>
 int flag=0;
%}

%token NUMBER

%left '+' '-'

%left '*' '/' '%'

%left '(' ')'

%%

ArithmeticExpression: E{

       printf("\nResult=%d\n", $$);

       return 0;

      };
 E:E'+'E {$$=$1+$3;}

 |E'-'E {$$=$1-$3;}

 |E'*'E {$$=$1*$3;}

 |E'/'E {$$=$1/$3;}
```

```
 |E'%'E {$$=$1%$3;}


 |'('E')' {$$=$2;}


 | NUMBER {$$=$1;}


 ;


%%


//driver code

void main()

{

  printf("\nEnter Any Arithmetic Expression which can have operations Addition, Subtraction,
Multiplication, Division, Modulus and Round brackets:\n");

  yyparse();

  if(flag==0)

  printf("\nEntered arithmetic expression is Valid\n\n");

}


void yyerror()

{

  printf("\nEntered arithmetic expression is Invalid\n\n");

  flag=1;

}
```

Output: -

```
Enter Any Arithmetic Expression which can have operations Addition, Subtraction, Multiplication, Division, Modulus and Round brackets:
(10*5*6)+20

Result=320

Entered arithmetic expression is Valid
```

```
Enter Any Arithmetic Expression which can have operations Addition, Subtraction, Multiplication, Division, Modulus and Round brackets:
(10*20)-100/2

Result=150

Entered arithmetic expression is Valid
```