LEX:

```
%{
#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;
}
```

Yacc:

```
%{
#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;}
 ;
%%
void main()
{
        printf("\nEnter Arithmetic Expression which can have operations Additiion,
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;}
```

```
student@user-H510M-H:~/Desktop/ARJUN/calculator$ ./a.out

Enter Arithmetic Expression which can have operations Additiion, Subtraction, Multiplication, Division,
Modulus and Round brackets:
(1+3)*(20)

Result=80

Entered arithmetic expression is Valid
```