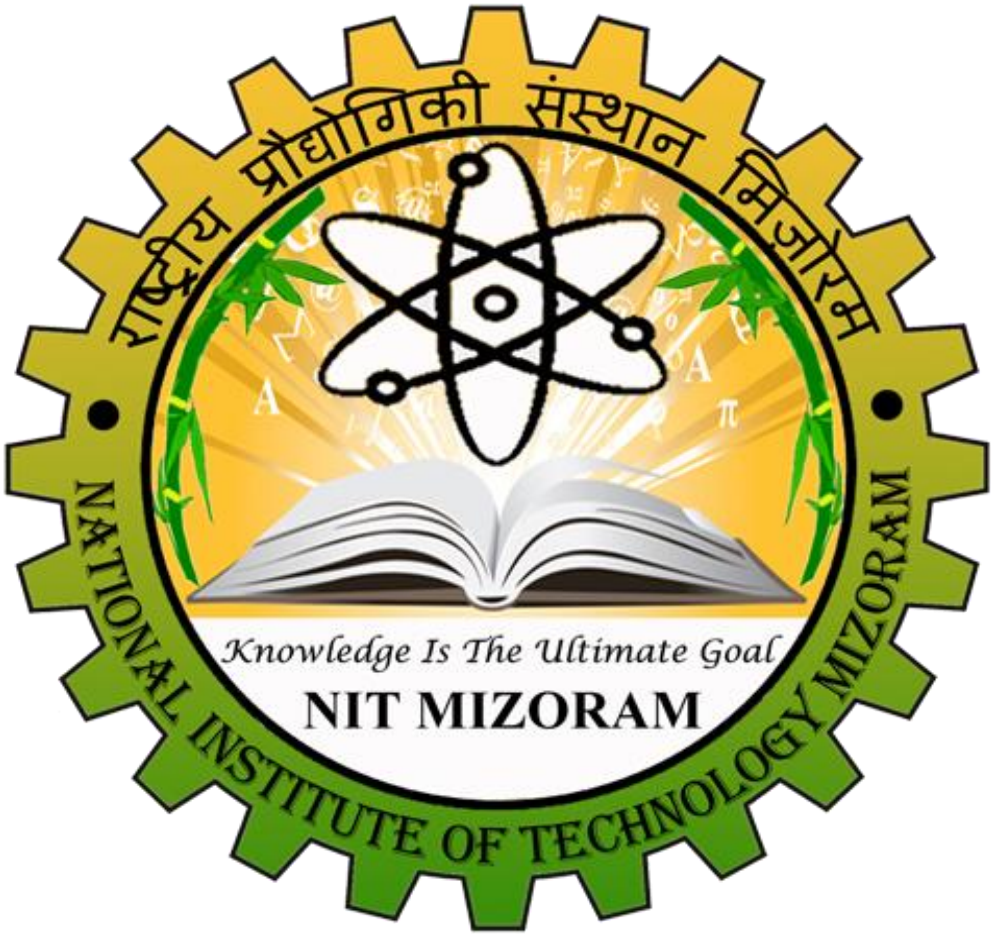
NATIONAL INSTITUTE OF TECHNOLOGY MIZORAM



COMPILER DESIGN LAB ASSIGNMENT

Name: NIRAJ KUMAR

Department: CSE

Enrollment No. : BT19CS031

6. Program to test the validity of a simple expression.

[Link to code at Github]

Lex Code

```
%{
#include"BT19CS031_Q6.tab.h"
%}
%%
[0-9]+ {return NUMBER;}
[a-zA-Z][a-zA-Z0-9_]* {return ID;}
\n {return NL;}
. {return yytext[0];}
%%
```

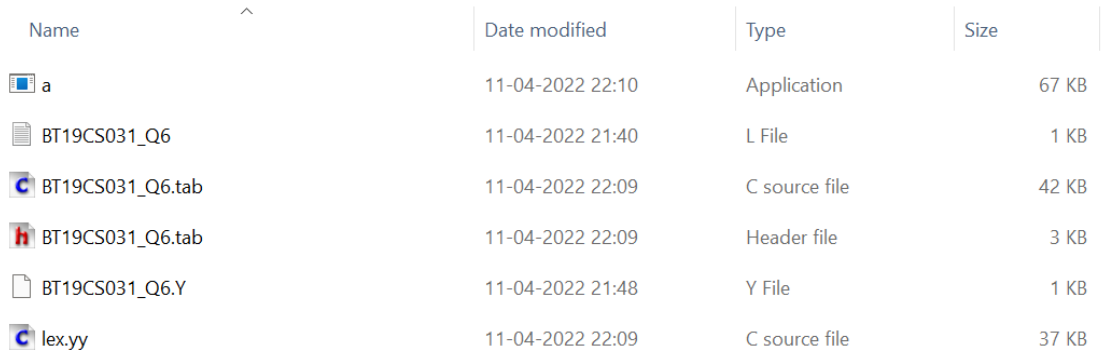
Yacc Code

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

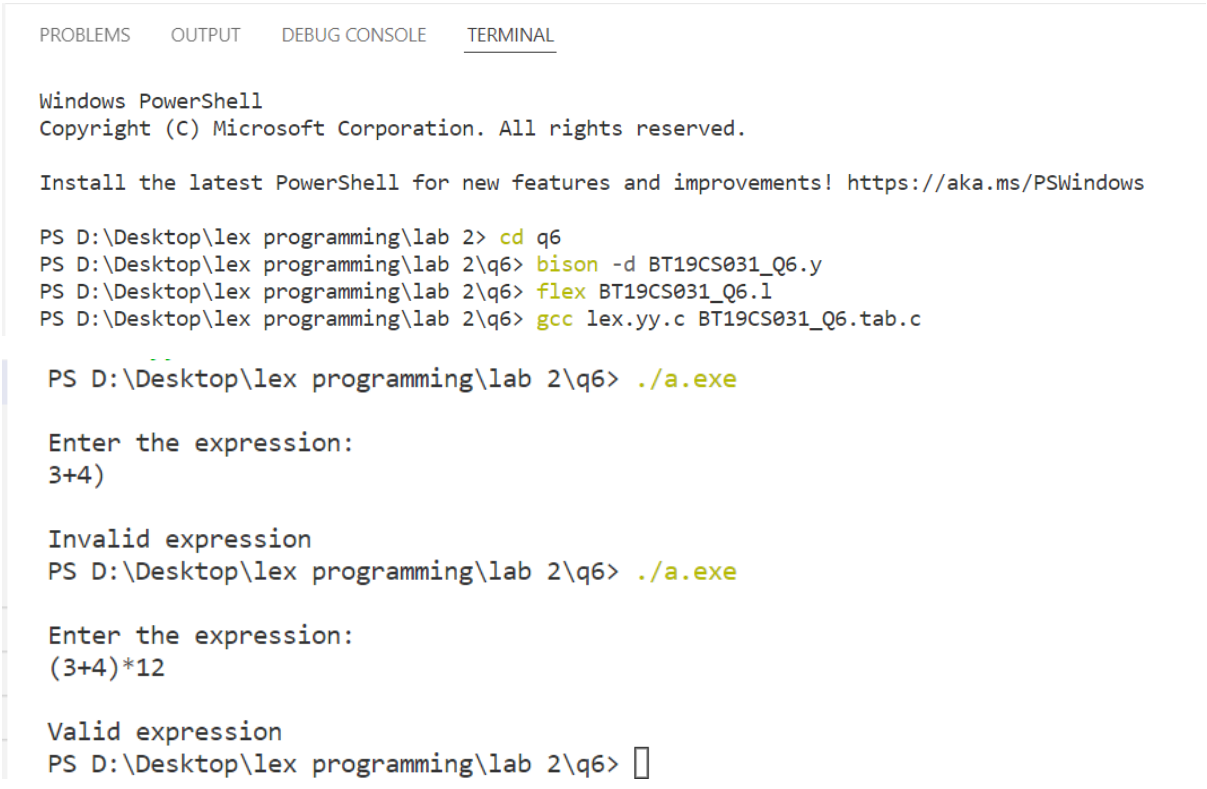
```
#include<stdlib.h>
%}
%token NUMBER ID NL
%left '+' '-' %left '*' '/'
%%
stmt: exp NL {printf("\nValid expression\n"); exit(0);}
;
exp: exp '+' exp | exp '-' exp | exp '*' exp | exp '/' exp | '(' exp ')' |
ID | NUMBER
;
%%
int yyerror(char *msg)
{
printf("\nInvalid expression\n");
exit(0);
}
int yywrap(){
return 1;
}
int main()
{
printf("\nEnter the expression: \n");
yyparse();
return 0;
```

```
}
```

OUTPUT



Name	Date modified	Type	Size
a	11-04-2022 22:10	Application	67 KB
BT19CS031_Q6	11-04-2022 21:40	L File	1 KB
BT19CS031_Q6.tab	11-04-2022 22:09	C source file	42 KB
BT19CS031_Q6.tab	11-04-2022 22:09	Header file	3 KB
BT19CS031_Q6.Y	11-04-2022 21:48	Y File	1 KB
lex.yy	11-04-2022 22:09	C source file	37 KB



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS D:\Desktop\lex programming\lab 2> cd q6
PS D:\Desktop\lex programming\lab 2\q6> bison -d BT19CS031_Q6.y
PS D:\Desktop\lex programming\lab 2\q6> flex BT19CS031_Q6.l
PS D:\Desktop\lex programming\lab 2\q6> gcc lex.yy.c BT19CS031_Q6.tab.c
PS D:\Desktop\lex programming\lab 2\q6> ./a.exe

Enter the expression:
3+4)

Invalid expression
PS D:\Desktop\lex programming\lab 2\q6> ./a.exe

Enter the expression:
(3+4)*12

Valid expression
PS D:\Desktop\lex programming\lab 2\q6>
```

7. Program to count the number of identifiers

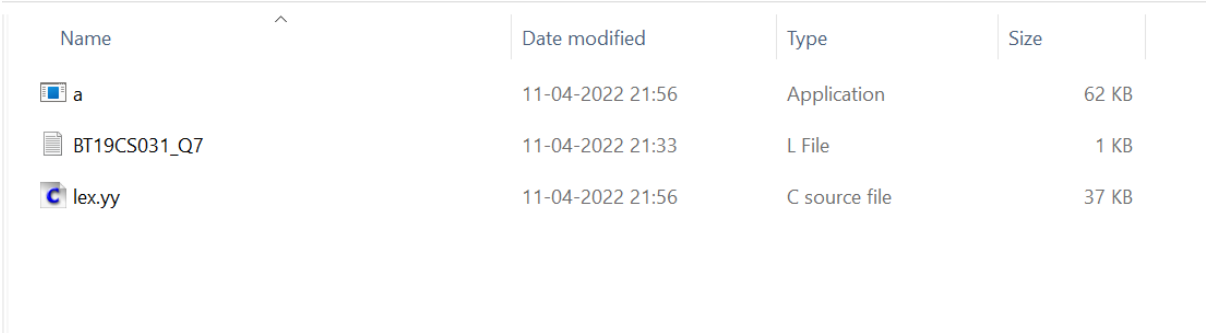
[Link to code at Github]

Lex Code

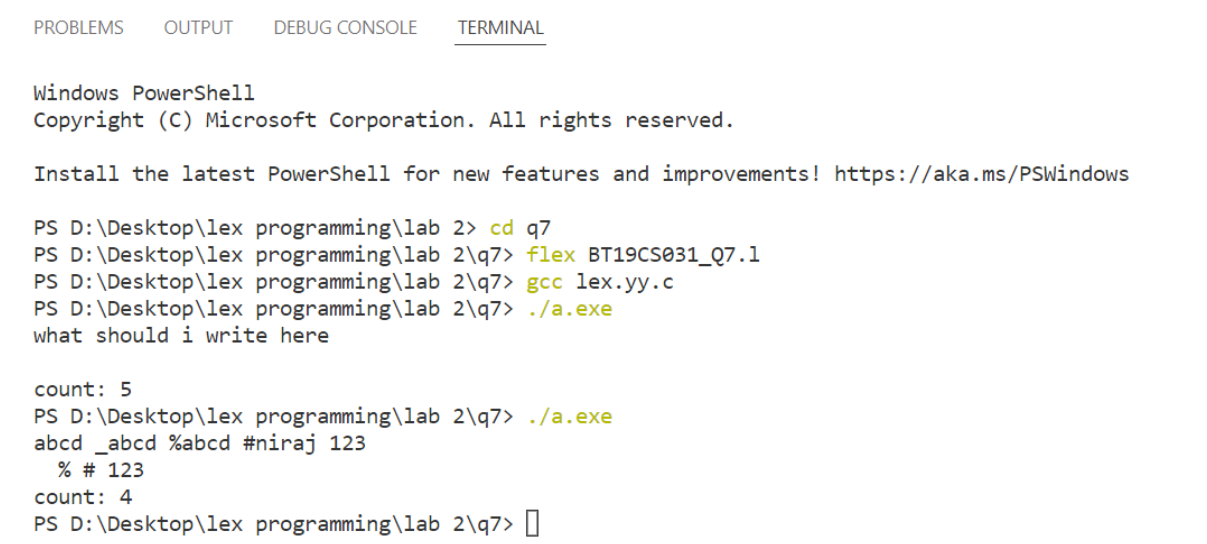
```
%{#include<iostream.h>
int count=0;
char ch=0;
%}
digit[0-9]
letter[a-zA-Z_]
%%
{letter}({letter}|{digit})* {
count++;
}
%%
int yywrap(){
return 1;
}
int main()
{
```

```
yylex();
printf("count: %d",count);
return 0;
}
```

OUTPUT



Name	Date modified	Type	Size
a	11-04-2022 21:56	Application	62 KB
BT19CS031_Q7	11-04-2022 21:33	L File	1 KB
lex.yy	11-04-2022 21:56	C source file	37 KB



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS D:\Desktop\lex programming\lab 2> cd q7
PS D:\Desktop\lex programming\lab 2\q7> flex BT19CS031_Q7.l
PS D:\Desktop\lex programming\lab 2\q7> gcc lex.yy.c
PS D:\Desktop\lex programming\lab 2\q7> ./a.exe
what should i write here

count: 5
PS D:\Desktop\lex programming\lab 2\q7> ./a.exe
abcd _abcd %abcd #niraj 123
  % # 123
count: 4
PS D:\Desktop\lex programming\lab 2\q7>
```

8. Program to recognize valid variables, which start with a letter, followed by any number of letters or digits.

[Link to code at Github]

Lex Code

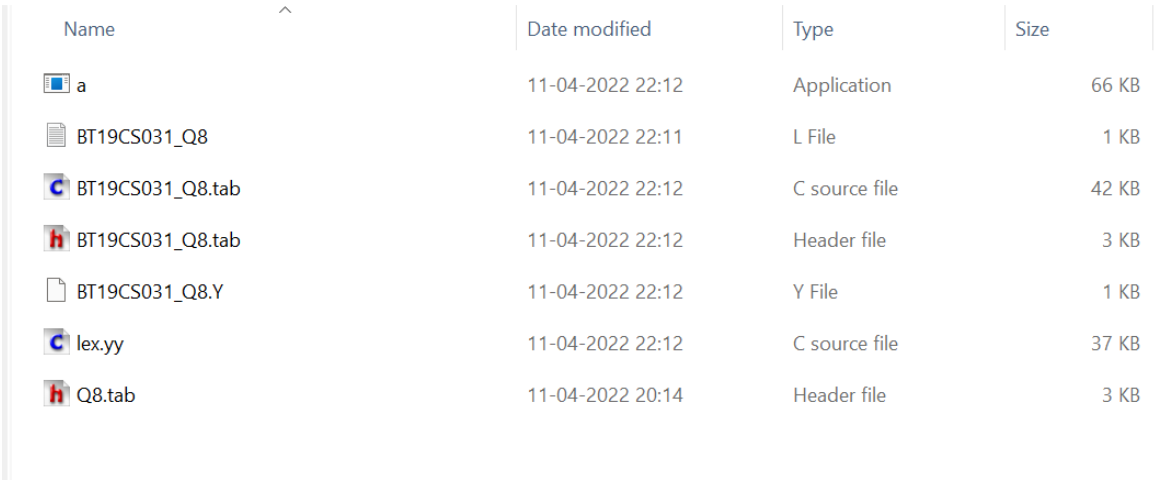
```
%{
 #include "BT19CS031_Q8.tab.h"
%}
%%
[a-zA-z_] {return ALPHA;}
[0-9]+ {return NUMBER;}
"\n" { return ENTER;}
. {return ER;}
%%
int yywrap()
{
return 1;
}
```

Yacc Code

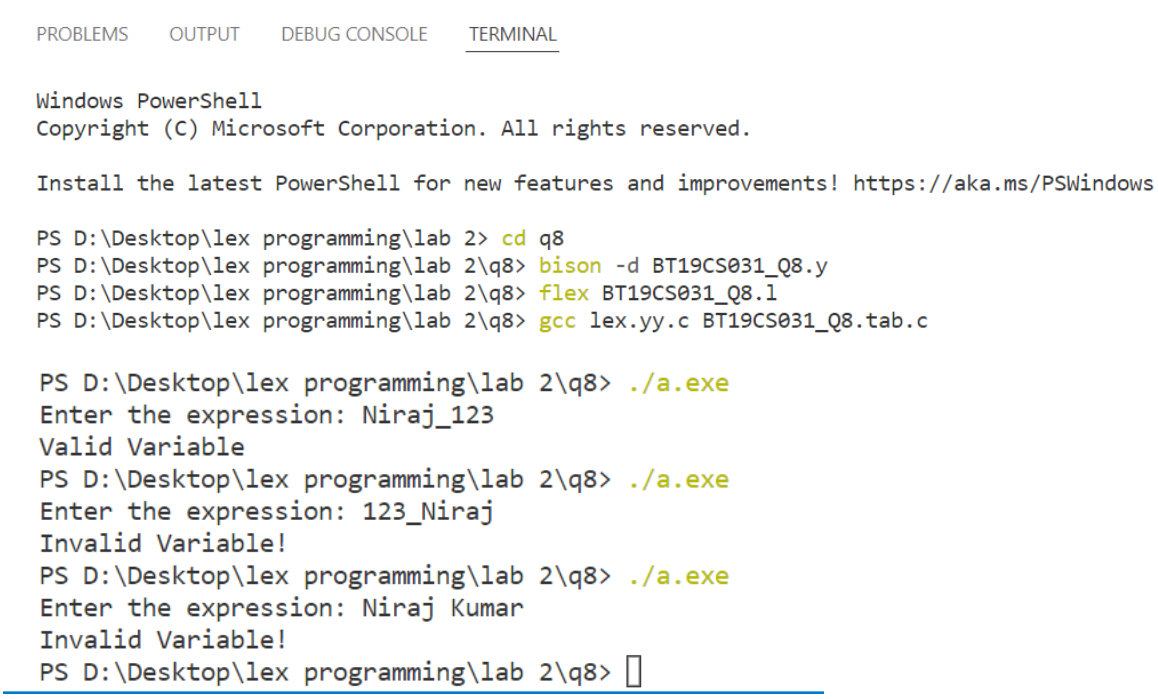
```
%{#include<stdio.h>
 #include<stdlib.h>
%}
%token ALPHA NUMBER ENTER ER
%%
var:v ENTER {printf("Valid Variable\n");exit(0);}
v:ALPHA exp1
exp1: ALPHA exp1
|NUMBER exp1
|;
%%
yyerror()
{
printf("Invalid Variable!");
}
int main()
{
printf("Enter the expression: ");
yyparse();
return 0;
```

}

OUTPUT



Name	Date modified	Type	Size
a	11-04-2022 22:12	Application	66 KB
BT19CS031_Q8	11-04-2022 22:11	L File	1 KB
BT19CS031_Q8.tab	11-04-2022 22:12	C source file	42 KB
BT19CS031_Q8.tab	11-04-2022 22:12	Header file	3 KB
BT19CS031_Q8.Y	11-04-2022 22:12	Y File	1 KB
lex.yy	11-04-2022 22:12	C source file	37 KB
Q8.tab	11-04-2022 20:14	Header file	3 KB



```
PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS D:\Desktop\lex programming\lab 2> cd q8
PS D:\Desktop\lex programming\lab 2\q8> bison -d BT19CS031_Q8.y
PS D:\Desktop\lex programming\lab 2\q8> flex BT19CS031_Q8.l
PS D:\Desktop\lex programming\lab 2\q8> gcc lex.yy.c BT19CS031_Q8.tab.c

PS D:\Desktop\lex programming\lab 2\q8> ./a.exe
Enter the expression: Niraj_123
Valid Variable
PS D:\Desktop\lex programming\lab 2\q8> ./a.exe
Enter the expression: 123_Niraj
Invalid Variable!
PS D:\Desktop\lex programming\lab 2\q8> ./a.exe
Enter the expression: Niraj Kumar
Invalid Variable!
PS D:\Desktop\lex programming\lab 2\q8>
```

9. Program to recognize the grammar a^nb^n, n<=20

[Link to code at Github]

Lex Code

```
%{
#include"BT19CS031_Q9.tab.h"
%}
%%
[a] return A;
[b] return B;
%%
```

Yacc Code

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

```
%token A B
%%
S:A S B
|
;
%%
int yywrap(){
return 1;
}
int main()
{
printf("Enter the string\n");
if(yyparse()==0)
{
printf("Valid string\n");
}
return 1;
}
yyerror(char *s)
{
printf("Invalid string\n");
}
```

OUTPUT

Name	Date modified	Type	Size
a	11-04-2022 22:16	Application	66 KB
BT19CS031_Q9	11-04-2022 22:15	L File	1 KB
BT19CS031_Q9.tab	11-04-2022 22:16	C source file	41 KB
BT19CS031_Q9.tab	11-04-2022 22:16	Header file	3 KB
BT19CS031_Q9.y	11-04-2022 22:15	Y File	1 KB
lex.yy	11-04-2022 22:16	C source file	37 KB

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS D:\Desktop\lex programming\lab 2> cd q9
PS D:\Desktop\lex programming\lab 2\q9> bison -d BT19CS031_Q9.y
PS D:\Desktop\lex programming\lab 2\q9> flex BT19CS031_Q9.l
PS D:\Desktop\lex programming\lab 2\q9> gcc lex.yy.c BT19CS031_Q9.tab.c

PS D:\Desktop\lex programming\lab 2\q9> ./a.exe
Enter the string
aaaabbbb

Valid string
PS D:\Desktop\lex programming\lab 2\q9> ./a.exe
Enter the string
bbbccc
Invalid string
PS D:\Desktop\lex programming\lab 2\q9> ./a.exe
Enter the string
aaabbbaaa
Invalid string
PS D:\Desktop\lex programming\lab 2\q9> ./a.exe
Enter the string
aaaaaaaaaabbbbbbbbbb

Valid string
PS D:\Desktop\lex programming\lab 2\q9>
```

10. Program to recognize whether a given sentences is simple or compound.

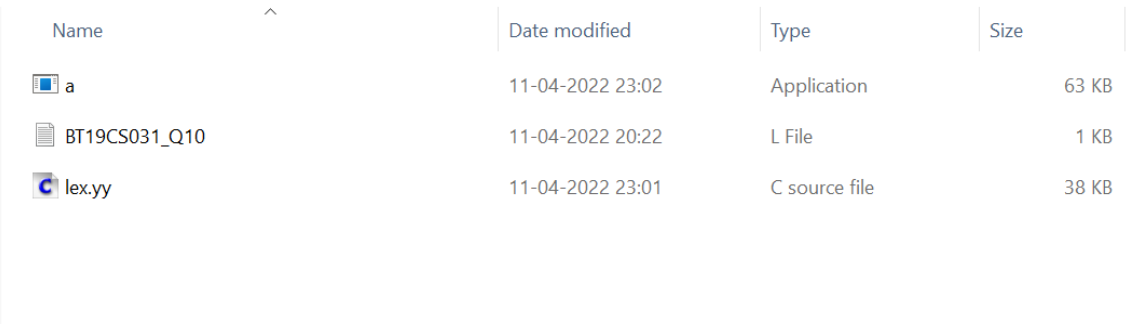
[Link to code at Github]

Lex Code

```
%{
int flag=0;
%}
%%
(" "[aA][nN][Dd]" ")|(" "[oO][Rr]" ")|(" "[bB][uU][tT]" ")|(" "[bB][Ee][Cc][Aa][Uu][Ss][Ee]" ") {flag++;}
.;
%%
int yywrap()
{
return 1;
}
int main()
{
printf("enter a sentence \n");
```

```
yylex();
if(flag==1)
printf("\ncompound sentence \n");
else
printf("\nsimple sentence \n");
return 0;
}
```

OUTPUT



Name	Date modified	Type	Size
a	11-04-2022 23:02	Application	63 KB
BT19CS031_Q10	11-04-2022 20:22	L File	1 KB
lex.yy	11-04-2022 23:01	C source file	38 KB

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS D:\Desktop\lex programming\lab 2> cd q10
PS D:\Desktop\lex programming\lab 2\q10> flex BT19CS031_Q10.l
PS D:\Desktop\lex programming\lab 2\q10> gcc lex.yy.c
PS D:\Desktop\lex programming\lab 2\q10> ./a.exe
enter a sentence
My namr is Niraj and I am from NIT Mizoroam
My namr is NirajI am from NIT Mizoroam

compound sentence
PS D:\Desktop\lex programming\lab 2\q10> ./a.exe
enter a sentence
Would you like to sit here or next to the window
Would you like to sit herenext to the window

compound sentence
PS D:\Desktop\lex programming\lab 2\q10> ./a.exe
enter a sentence
A day without sunshine is like night
A day without sunshine is like night

simple sentence
PS D:\Desktop\lex programming\lab 2\q10> ./a.exe
enter a sentence
Reality continues to ruin my life
Reality continues to ruin my life

simple sentence
PS D:\Desktop\lex programming\lab 2\q10>
```

11. Program to implement calculator using LEX/YACC.

[Link to code at Github]

Lex Code

```
%{
#include<stdio.h>
#include "BT19CS031_Q11.tab.h"
extern int yylval;
%}
%%
[0-9]+ {
yylval=atoi(yytext);
return NUMBER;}
[\t] ;
[\n] return 0;
. return yytext[0];
%%
int yywrap()
{
return 1;
```

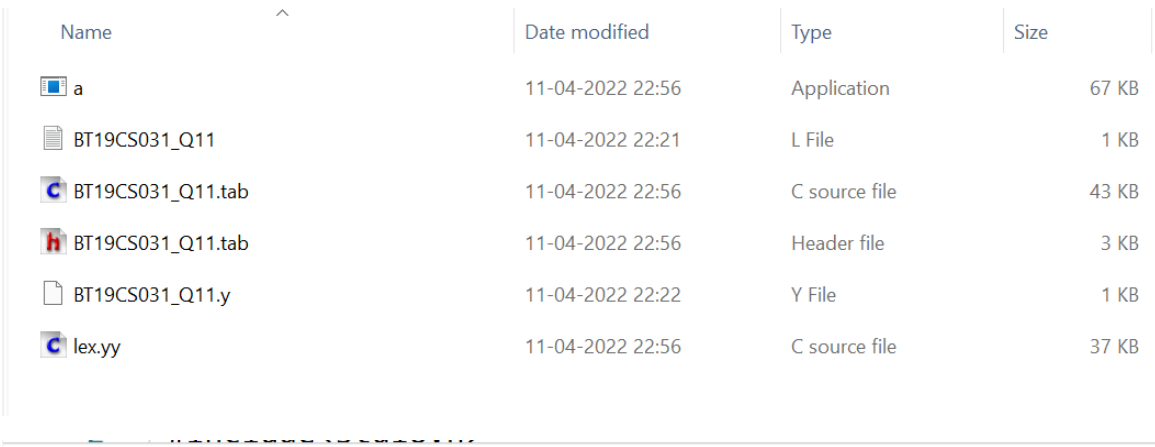
```
}
```

Yacc 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;}
;
%%
yyerror()
{
if(flag==1)
{
 printf("\nEntered arithmetic expression is Invalid\n\n");
}
}
int main()
{
printf("\nEnter Any Arithmetic Expression which can have operations Addition, Subtraction, Multiplication, Divison, Modulus and Round brackets:\n");
 yyparse();
 if(flag==0)
{
 printf("\nEntered arithmetic expression is Valid\n\n");
}
return 0;
}
```

OUTPUT



Name	Date modified	Type	Size
a	11-04-2022 22:56	Application	67 KB
BT19CS031_Q11	11-04-2022 22:21	L File	1 KB
BT19CS031_Q11.tab	11-04-2022 22:56	C source file	43 KB
BT19CS031_Q11.tab	11-04-2022 22:56	Header file	3 KB
BT19CS031_Q11.y	11-04-2022 22:22	Y File	1 KB
lex.yy	11-04-2022 22:56	C source file	37 KB

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS D:\Desktop\lex programming\lab 2> cd q11
PS D:\Desktop\lex programming\lab 2\q11> bison -d BT19CS031_Q11.y
PS D:\Desktop\lex programming\lab 2\q11> flex BT19CS031_Q11.l
PS D:\Desktop\lex programming\lab 2\q11> gcc lex.yy.c BT19CS031_Q11.tab.c
```

```
PS D:\Desktop\lex programming\lab 2\q11> ./a.exe

Enter Any Arithmetic Expression which can have operations Addition, Subtraction, Multiplication, Divison, Modulus and Round brackets:
(1+2+3)*(3+4+5)/(1+2)

Result=24

Entered arithmetic expression is Valid

PS D:\Desktop\lex programming\lab 2\q11> ./a.exe

Enter Any Arithmetic Expression which can have operations Addition, Subtraction, Multiplication, Divison, Modulus and Round brackets:
((23+27)*(33+37))%11

Result=2

Entered arithmetic expression is Valid

PS D:\Desktop\lex programming\lab 2\q11>
```