

## Lab Practice Session # 4

Course Title: Compiler Construction Lab (CSTE-4106)

1. Write a C program to generate three address code.

Sample Input	Sample Output
a = b + c + d - e	t1=b+c t2=t1+d t3=t2-e a=t3

2. Follow this link to download Bison (Yacc): [How to Compile Run LEX YACC Programs on Windows - VTUPulse](#). Then run the following code:

### Yacc Part

```
%token NUMBER ID NL
%left '+' '-'
%left '*' '/'
%%
stmt : exp NL { printf("Valid Expression"); exit(0); }
;
exp : exp '+' exp
    | exp '-' exp
    | exp '*' exp
    | exp '/' exp
    | '(' exp ')'
    | ID
    | NUMBER
;
%%
int yyerror(char *msg)
{
    printf("Invalid Expression\n");
    exit(0);
}
main ()
{
    printf("Enter the expression\n");
    yyparse();
}
```

### Lex Part

```
% {
#include "y.tab.h"
% }
%%
[0-9]+ { return DIGIT; }
```

```
[a-zA-Z][a-zA-Z0-9_]* { return ID; }
\n { return NL ;}
.
{ return yytext[0]; }
%%
```

## Compiling and Running YACC Programs:

Use the following commands to compile and run the **YACC** program in the Windows environment.

**(Note: hello.l is the LEX and hello.y is the program)**

```
flex hello.l
```

```
bison -dy hello.y
```

```
gcc lex.yy.c y.tab.c
```

```
a.exe
```

### Assignment (Report #3)

1. Write a C program to implement operator precedence parsing.

Sample Input	Sample Output
(d*d)+d\$	Precedence input: \$<(<d>*<d>)>+<d>\$ \$<(E*<d>)>+<d>\$ \$<(E*E)>+<E>\$ \$E+E\$ Precedence Input: \$<+>\$ \$E\$ Success

2. Write a YACC Program to recognize a valid variable, which starts with a letter, followed by any number of letters or digits.
3. Write a YACC Program to recognize nested IF control statements and display the levels of nesting.

**Submission Deadline: 07/10/2024**