

Experiment 5

SUBJECT: COMPILER DESIGN LAB

Experiment : Conversion of Infix to Post fix using Lex and Yacc

YACC (Yet another Compiler-Compiler) is the standard parser generator for the Unix operating system. An open source program, yacc generates code for the parser in the C programming language. The acronym is usually rendered in lowercase but is occasionally seen as YACC or Yacc.

ALGORITHM:

1. Scan all the symbols one by one from left to right in the given Infix Expression.
2. If the reading symbol is operand, then immediately append it to the Postfix Expression .
3. If the reading symbol is left parenthesis '(', then Push it onto the Stack.
4. If the reading symbol is right parenthesis ')', then Pop all the contents of the stack until the respective left parenthesis is popped and append each popped symbol to Postfix Expression.
5. If the reading symbol is operator (+ , - , * , /), then Push it onto the Stack. However, first pop the operators which are already on the stack that have higher or equal precedence than the current operator and append them to the postfix . If open parenthesis is there on top of the stack then push the operator into the stack.
6. If the input is over, pop all the remaining symbols from the stack and append them to the postfix

CODE:

LEX CODE:

```
%{
#include"y.tab.h"
extern int yylval;
}%
%%
[0-9]+ {yylval=atoi(yytext); return NUM;}
\n    return 0;
.      return *yytext;
%%

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

YACC CODE:

```
%{
#include<stdio.h>
}%
%token NUM
%left '+' '-'
%left '*' '/'
%right NEGATIVE
%%
S: E {printf("\n");}
;
E: E '+' E {printf("+");}
| E " E {printf("*");}
| E '-' E {printf("-");}
| E '/' E {printf("/");}
| '(' E ')'
```

```

| '-' E %prec NEGATIVE {printf("-");}
| NUM    {printf("%d", yylval);}
;
%%

int main(){
  yyparse();
}

int yyerror (char *msg) {
  return printf ("error YACC: %s\n", msg);
}

```

COMMANDS:

```

gedit ip.l
ctrl c
gedit ip.y
ctrl c
yacc -d ip.y
flex ip.l
cc lex.yy.c y.tab.c
./a.out

```

OUTPUT:

```
ex1@AB1210SCOPE70: ~
File Edit View Search Terminal Help
^C
ex1@AB1210SCOPE70:~$ yacc -d ip.y
ex1@AB1210SCOPE70:~$ flex ip.l
ex1@AB1210SCOPE70:~$ cc lex.yy.c y.tab.c
y.tab.c: In function 'yyparse':
y.tab.c:1219:16: warning: implicit declaration of function 'yylex' [-Wimplicit-f
unction-declaration]
 1219 |         yychar = yylex ();
      |                   ^~~~~~
y.tab.c:1388:7: warning: implicit declaration of function 'yyerror'; did you mea
n 'yyerrok'? [-Wimplicit-function-declaration]
 1388 |         yyerror (YY_("syntax error"));
      |         ^~~~~~
      |         yyerrok
ex1@AB1210SCOPE70:~$ ./a.out
2+6*2-5/3
262*+53/-
ex1@AB1210SCOPE70:~$ gedit ip.l
^C
ex1@AB1210SCOPE70:~$ gedit ip.l
^C
ex1@AB1210SCOPE70:~$ gedit ip.y
^C
ex1@AB1210SCOPE70:~$
```