Expt. No Page No4
Expt. No. 2 Page No. 4 Expt. Name. Use LEX and YACC to implement parser on ambigous Date:
A `
Aim:
To write the program using LEX and YACC to implement parser on ambigous grammar.
on ambigues grammar
B Algorithm: > File.1
Step 1: Start
step 2: Include the necessary header files and declare the
MPC PHONE MANIA LIA
step 3: initialize the digits, operators, parenthesis and return the
value else print accordo Syntax Error
step 4: Call the function & return I
Step 5: Stop
File. y step1: Start
step1: Start step2: Include the necessary header files and declare the
necessary variables
step 3: Substitute the values and Calculate respectively for
Addition, Subtraction, Multipalication and division and
return the result.
step 4: Call the main function and print the result
steps: stop.

Expt. No	Page No5
Expt. Name	Date :
Program:	
File !	
Y. option noyywrap	
7. &	
#include < stdio. h>	
#include "y.tab.h"	
void yyerror (chor #s)	
extern int yylval;	
7.3	
[0-9] + Eyylval = a toi(yytext);	
return NUM;	
[-+*/\n] {return *yytext;}	
"C" Ereturn*yytext;}	
">" { return *yy text)}	
rit];	
¿yyerror ("Syntax Error");}	
<u> </u>	
int yywrap()	
8	
return 1;	
4	
File.y	The second description of the second party of the second second is a second second second second second second
78	entitytein tellitytein tellityin tellitytein tellityin
#include < stdio.h>	
extern Int yylex (void);	
extern Int yylex (void); void yyerror (char *);	
7. 3	

Expt. No.	Page No6
Expt. Name	Date :
7. token NUM	
y. %.	
5:	
S expr 'm' &printf ("xd In", \$2); 3	
·	
expr:	
expr. + expr {\$\$ = \$1 + 43}	
lexpr'-'expr &\$\$=\$1#\$3:}	
1 expr * expr \$\$\$ = \$1* \$3.5	
1 expr 1' expr	
1 NUM 2\$\$ = \$1.3	
1 '('expr')' { \$\$ = \$2;}	
Y. Y.	
void yyerbor (char *s)	
4	
printf ("xsln",s);	
'g	
int main()	
.	
yy porse();	
return 0; 3	
Result:	
Use LEX and YACC to implement parser for a	imbiguous is
executed successfully.	
AND	