# VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.



### **COMPILER DESIGN**

### **Submitted by**

JYOTHIKA C N (1BM21CS083)

in partial fulfillment for the award of the degree of BACHELOR OF ENGINEERING in COMPUTER SCIENCE AND ENGINEERING



B.M.S. COLLEGE OF ENGINEERING (Autonomous Institution under VTU) BENGALURU-560019 Oct 2023-Feb 2024

B. M. S. College of Engineering,

## Bull Temple Road, Bangalore 560019 (Affiliated To Visvesvaraya Technological University, Belgaum) Department of Computer Science and Engineering



#### **CERTIFICATE**

This is to certify that the Lab work entitled "COMPILER DESIGN" carried out by JYOTHIKA C N(1BM21CS083), who is bonafide student of B. M. S. College of Engineering. It is in partial fulfillment for the award of Bachelor of Engineering in Computer Science and Engineering of the Visvesvaraya Technological University, Belgaum during the year 2022-23. The Lab report has been approved as it satisfies the academic requirements in respect of Compiler Design Lab - (22CS5PCCPD) work prescribed for the said degree.

#### Sunayana S

Assistant Professor Department of CSE BMSCE, Bengaluru Dr. Jyothi S Nayak

Professor and Head Department of CSE BMSCE, Bengaluru



NAME dyothika. CN SEC. 5B ROLL NO. CD LAB s, No. Date Title Sign / 110. 20/11/2023 Court the number z voucle and corecnerts 20/11/2023 Identify tokens, keywords A 11/2023 Floating point numbers Replacing sequence of non-empty spaces, with single spaces 4/12/2023 4 11/12/2023 Recognize tokens over alphabets 40.893 18/12/2023 program to design 11/1/2024 Recursive descent Derk Calculator 11/1/2024 11/1/2024 String parter Syntax tree generator 29/1/2024 Topix to postfix ming 29/1/2024 Three -address code generator wing the 29/1/2024

```
Write a lex program to identify each character as
compared or voud in a given sentence and also
court the vowel and commonants
                                          (Red Struck
 1. option noyyurap
 #include <stdio. h>
 int 120;
 int (=0;
  1.3
  1.7.
  [aciouAEIOU] fV++; print ("vowel: x.SIn; yytext); }
  [a-2 A-Z] fett; print (" comon ant: 1.5 h, yyter))
  [ In] & print C'incernber of would it In number of
                 Commands 1.d" U, C); y
  1.7.
  int main 4
   yylex17.
     retur o
autput
 airug
  Voud: a
  wwel: i
  conconant: ~
 Warel:4
```

Convonant 19

number of vowels 3

number 9 consonants 2

```
pmscecse@bmscecse-OptiPlex-5070:-/Documents/1BM21CS083$ lex p4.l
omscecse@bmscecse-OptiPlex-5070:~/Documents/18M21C5083$ gcc lex.yy.c
omscecse@bmscecse-OptiPlex-5070:-/Documents/18M21CS083$ ./a.out
abcdef
owel:a
consonant:b
consonant:c
consonant:d
owel:e
consonant:f
number of vowels 2
number of consonants 4
```

valid token on the terminal Write a lex program to frame [20]; int | Ploat | char Sprint ("Reywords"). 5
[a-z A-z]\* Sprint ("Identifies "); 5
); Sprint ("Seperator"); 5

```
yyurapi,
 main ()
painty ( enter the japant (it name (n").
 s carp ("1. 3", frame),
points ("enter the output fil nameln"),
 sanbl" 1.5", forame),
yours = Ropen (frame, "x");
 yyout = topen (foram, "w"),
yylex ();
 flor (gyin)
  felire (yy, n):
```

bmscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21CS083\$ lex p.l bmscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21C5083\$ qcc lex.vv.c bmscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21CS083\$ ./a.out enter the input file name enter the output file name output.txt

mscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21CS083S

input.txt

1 int a,b;

int Keywords a Identifiers, Seperatorb Identifiers; Seperator

```
Floating point number
29/11/2023
Write a program in LEX to sucognize
check for all the following input cases
 7.5
  4.3
  [+-]? [0-9]*[.][0-9]+ Sprint ("floating point numbers lay)
  7.4.
  [+-]?[0-9]+ [pnnt] ("not a gloating point purnbulny), y
   int yyurapi)
   void main()
       pmh +6 ("enter any number");
      yylex 1),
  Output
 enter any number 23.6
   floating point numbers
  45
  not a floating point number
  +6.3
  floating point number
  - 55.66
   floating point number
   55.
    not a floating point number
   .33
```

floating point number

```
omscecse@bmscecse-OptlPlex-5070:~/Documents/1BM21CS083$ lex float.l
omscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21C50B3$ gcc lex.yy.c
pmscecse@bmscecse-OptiPlex-5070:~/Documents/18M21C5083$ ./a.out
enter any number 23.6
floating point numbers
not a floating point number
+6.3
floating point numbers
-55.66
floating point numbers
55.
not a floating point number
```

```
Write a LEX program that copies a file, replacing
      nonempty sequence of white spaces by a single
each
blank.
       the east in a let realed experient
  1.5
   #tindude < stdio. h>
   #include & string. h>
   #induderstdlib. h>
   chan strillaso].
  [10] floor (yyout, x, s\n", stri); stri [0] = 101; 3
   [ ]* [14] fprint (yyout, "1.5", str); str PO] = 10'
  strict (str), yetext);

(FOF>> Sprint (yyout, "x.s", str); return 0; 3
  int main ()
       entern FIE xyyin, xyyoud;
      char filename [100];
      print ( Enter the name of the Ale to copy: 1t").
       stand ( 4 x s , filename).
        yyin, topen (filenam, """): " of a got a ago
        of Cygin == NULLS ("W" [] upon) angel a tange
       print (4 Enter the name of the Bile to work 1 to);
        stand ( " y. s", filename).
       yyout = topen (filenam, "w").
       if (gyout = 2 Nows)
           exit (1)
       yylex ();
     int yywrap (void)
```

```
bmscecse@bmscecse-OptiPlex-5070: ~/Documents/1BM21CS...
 F
                                                           Q =
bmscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21CS083$ cc lex.yy.c
bmscecse@bmscecse-OptiPlex-5070:-/Documents/1BM21CS083$ ./a.out
9000
success
bmscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21CS083$ ./a.out
4005
success
bmscecse@bmscecse-OptiPlex-5070:~/Documents/18M21C5083$ ./a.out
123
123fail
bmscecse@bmscecse-OptiPlex-5070:~/Documents/18M21CS083$ lex re7.l
bmscecse@bmscecse-OptiPlex-5070:-/Documents/1BM21CS083$ cc lex.yy.c
bmscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21CS083$ ./a.out
1234
success
bmscecse@bmscecse-OptiPlex-5070:~/Documents/18M21C5083$ ./a.out
4511
fail
omscecse@bmscecse-OptiPlex-5070:~/Documents/18M21CS083$ lex blank.l
omscecse@bmscecse-OptiPlex-5070:~/Documents/18M21CS083$ cc lex.yy.c
omscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21C5083$ ./a.out
Enter the name of the file to copy:
                                       input.txt
Enter the name of the file to write: output.txt
omscecse@bmscecse-OptiPlex-5070:-/Documents/1BM21CS
```

re1.l	re2.l		re3.l	re4.l	re5.l	re6.l	re7.l	output.txt	input.txt	
python is an		programming								

```
(1)
   digits [0-9]
   adigits you aprint ("yes string ends with 00" yeters); y
   Edigits3* of printy ("yes string does not end with 60", yylens),
  Output
   12300
    12500 thing ends with or
(i) 7.7.
    Edigits 3+ 222 8 digits 3 * Sprint ("/s ship has 222; yeterd),3
    Edigitsy*
                   I paint ("yes string does not have 222" yesterb),
    Output
    122234
    122234 iting has 221
(10)
    thindude <math.h>
int value = 0, i,j =0, Hag= 0;
y. 3
    7.7.
            of for (i=gyleng-1; i>=0, i--)
                      value += (yetext(i)_48) *pow (2,j);
                  if (value 1. 5 = 20)
                  { lay=1;
           retur o;
    [in]
   1/2
  Oletput
   101
```

Success

sdigits )\* 1 sdigits 19 sprints (1xs 10th symbol from right end is 1 "yytext ); 5 Edigits3\* Epninth ("Y.s nots", yytex0) 3 1.7. Output 1023002245 1023002245 10th symbol from night end is 1 (vi) 1.1. f for (i-yyleng-1; i >=0; i--) Valuet = Cystext [17 -487, [in) returno, 1. 4 autput 4005 Succes (VD) fdigits )4 for (iso; ix yylens; it+) ib (yytext (i)> yytext(i+1)) f flag=0;

```
bmscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21CS083S ./a.out
1111
successbmscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21CS083$ ./a.out
11
bmscecse@bmscecse-OptiPlex-5070:-/Documents/1BM21CS083$ lex re5.l
bmscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21CS083$ cc lex.yy.c
bmscecse@bmscecse-OptiPlex-5070:~/Documents/18M21C5083S
bmscecse@bmscecse-OptiPlex-5070:-/Documents/1BM21CS083$ ./a.out
1023002245
1023002245 10th symbol from right end id 1
                              ./a.out
[1]+ Stopped
bmscecse@bmscecse-OptiPlex-5070:-/Documents/1BM21CS083$ lex re6.l
bmscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21CS083$ cc lex.yy.c
bmscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21CS083S ./a.out
9000
success
bmscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21C5083$ ./a.out
4005
success
bmscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21CS083$ ./a.out
123
123fail
     other and play some /newspace /spublicsoms lev re7 1
```

```
fail
bmscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21C5083$ lex blank.l
bmscecse@bmscecse-OptiPlex-5070:~/Documents/18M21CS083$ cc lex.yy.c
bmscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21CS083$ ./a.out
Enter the name of the file to copy: input.txt
Enter the name of the file to write: output.txt
bmscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21CS083$ lex re1.l
bmscecse@bmscecse-OptiPlex-5070:~/Documents/1BMZ1CS0B3$ cc lex.yy.c
bmscecse@bmscecse-OptiPlex-5070:~/Documents/18M21C5083$ ./a.out
24900
24900 string ends with 00
2352
2352 string does not end with 00
^Z
[2]+ Stopped
                              ./a.out
bmscecse@bmscecse-OptiPlex-5070:-/Documents/18M21CS083$ lex re2.l
bmscecse@bmscecse-OptiPlex-5070:-/Documents/1BM21C5083$ cc lex.yy.c
bmscecse@bmscecse-OptiPlex-5070:-/Documents/1BM21CS083$ ./a.out
12142
12142 string does not have 222
24322245
24322245 string has 222
```

```
mscecse@bmscecse-OptiPlex-5070:~/Documents/18M21C5083$ lex re4.l
mscecse@bmscecse-OptiPlex-5070:~/Documents/18M21CS083$ cc lex.vv.c
usr/bin/ld: /tmp/ccNpRHPT.o: in function `yylex':
ex.yy.c:(.text+0x33f): undefined reference to `pow'
ollect2: error: ld returned 1 exit status
mscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21CS083$ cc lex.vv.c -lm
mscecse@bmscecse-OptiPlex-5070:-/Documents/18M21CS083$ ./a.out
uccessbmscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21CS083$ cc lex.vv.c -lm
mscecse@bmscecse-OptiPlex-5070:-/Documents/1BM21CS083$ ./a.out
111
uccessbmscecse@bmscecse-OptiPlex-5070:-/Documents/18M21CS083$ ./a.out
mscecse@bmscecse-OptiPlex-5070:-/Documents/1BM21CS083$ lex re5.l
mscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21CS083$ cc lex.yy.c
mscecse@bmscecse-OptiPlex-5070:-/Documents/1BM21CS083S
mscecse@bmscecse-OptiPlex-5070:-/Documents/1BM21CS083S ./a.out
023002245
023002245 10th symbol from right end id 1
1]+ Stopped
                             ./a.out
mscecse@bmscecse-OptiPlex-5070:~/Documents/18M21CS0B3$ lex re6.l
mscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21C50B3$ cc lex.vv.c
mscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21C5083$ ./a.out
```

```
bmscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21CS083S lex re7.l
bmscecse@bmscecse-OptiPlex-5070:-/Documents/18M21C5083$ gcc lex.vv.c
bmscecse@bmscecse-OptiPlex-5070:-/Documents/1BM21CS083$ ./a.out
45612
2fail
bmscecse@bmscecse-OptiPlex-5070:-/Documents/1BM21C5083$ ./a.out
1234
success
```

Write a program to design lexical analyzer in c/c++/ Java/Python language (to recognize any five keywords, identifiers, numbers, operators, and punctuations) #inlude < stdio. ho #include < string. h> #include < ctype. h> void lexical Analyzon (chan input code (3) char \*keywords [] = f "ib", "else", "while", "for", "seturn", chan \*operators[] = 9"+", "-", " \* "<= " >= ""=3; char \* punctuation, [] = f",", ";"("")", "5", "33, chan \*tokens = statok (input-code, whole ( token != NULL) ib (isdigit (token [07)) print ("Number: 19/10", token); else if (isalpha(token(0)) 11 token(0) == = =) int iskeyword = 0; forlint i = 0% is size of (heywords) / size of (heywords [0]); 16(streen p (token, keywords[i] == 0) f print (" keyword: y.sln" token);
is keyword = 1; break;
3 (! iskeyword) printle " Identifier: Y.slo" token), else if (strehr ("+-\*/= <>; (),!", token (0))!=NULL) printle "Punctuation Operator: Y.Sin", token); token = strtok (NULL, " ItIn"); //next token

int main() chan input - code (2007; printle ("enter a code (n"); fgets (input code, 200, stdin) lexical Analyzer (input\_code); return o; Output enter coode int a = 1234) Keyword : int Identifier: a Punchuation Openations . =

Number: 1234 Punctipation/operator:

```
enter c code
int a = 1234;
Keyword: int
Identifier: a
Punctuation/Operator: =
Number: 1234
Punctuation/Operator:;
```

```
white a program to perform Recursive descent on the following
 grammon
                  S-CAd,
                  A -abla
 Hindude (stdio. h)
 Hindude < stellib. ho
                                     profit Taring
  char input [100].
  int ind = 0;
  void match (char expected)
  if (input (ind) == expected)
          ind+
    wid An;
    woid si)
         match ('c');
         match ('d')
      word A();
       wid 317
           match ('c').
       void A()
           if (inputsind] = 2 'a')
              printy ("Helbiny);
              match ('a').
              match ('b'),
           che
              proff ("Pavering failed. In", ind),
              exit(i).
           4
```

printf ("ther he input iting : \n"). scanf ("/s" input) Electron stellers h succenfull. \n') punt ("Parsing failed Extra characters found in "), seture of Output Enter the input iting cabd \$ Hello Parking Successful. (12, ) 42 + 1200

Christof father in ich

```
ecursive_descent.c: In function 'A':
ecursive descent.c:33:16: warning: too many arguments for format [-Wformat-extra-args]
               printf("Parsing failed.\n", ind);
mscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/DocumentsS ^C
mscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/Documents$ ^C
mscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:-/Documents$ gcc -o recursive descent recursive descent.c
ecursive descent.c: In function 'A':
ecursive_descent.c:33:16: warning: too many arguments for format [-Wformat-extra-args]
               printf("Parsing failed.\n", ind);
 33 I
mscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:-/Documents$ ./recursive descent
nter the input string:
ad
ello
arsing failed. Extra characters found.
mscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/Documents$ ./recursive descent
nter the input string:
aaad
ello
arsing failed. Extra characters found.
mscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:-/Documents$ ./recursive descent
nter the input string:
abs
ello
arsing successful.
mscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:-/Documents$ ./recursive descent
nter the input string:
aadS
ello
arsing failed. Extra characters found.
mscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:-/Documents$ ./recursive descent
nter the input string:
abd$
ello
arsing successful.
mscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:-/Documents$ ./recursive descent
nter the input string:
aaadS
ello
arsing failed. Extra characters found.
mscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/Documents$
```

```
Write a yack program to parce things
grammar L a'b n>5
 anbn.1
  1.8
 Hindude <Adio. h>
 Hindude Stellib. h>
#include "y tab. h"
extern int yyval;
[aA] Syphal system [0]; return A; 3
 [6B] Syral = gytext (0]: return B; 3
In Fretur NL; 3
      Fretuer yetext [0]; }
   return !
anbn.y
 #indude stdio. h>
 #indude (Adlib.h)
 int gyoner (cha *s).
 int yyux (void).
 1. token A
Xtoken B
Y. token NL
 Smit : AAAAASBNL Sprinty ("Parsed wing the rule (a16))
            n>= 5, In laid Shing !In"); 3
 S: SA
```

printly ("Enter a string!\n").

Hypanse ();

Int yyerror (char \*s)

printle ("Invalid string!\n"),

return o.

Output

\$ lex and o.!

\$ yacc -d and o.y

\$ gcc hexyy. c y.tab.c

\$ .la.out

Enter a string!

aaaaab

Pared ming the rule (arn)b, n>=5

Valid string!

aabb

trualid stringt

CHAILLS

(pion) with

mus redot

lept 1411

14. Allan

```
omscecse@bmscecse-OptiPlex-5070:~/Documents/18M21C5083S vacc -d anbn.v
omscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21C5083$ gcc lex.yy.c y.tab.c
mscecse@bmscecse-OptiPlex-5070:-/Documents/1BM21C5083$ ./a.out
inter a string!
labb$
nvalid String!
mscecse@bmscecse-OptiPlex-5070:-/Documents/1BM21CS0835 ./a.out
nter a string!
abb
nvalid String!
mscecse@bmscecse-OptiPlex-5070:-/Documents/18M21C50835 ./a.out
nter a string!
aaab
nvalid String!
mscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21CS083$ ./a.out
nter a string!
aaaab
arsed using the rule (a^n)b, n>=5.
alid String!
aaaaabb
nvalid String!
mscecse@bmscecse-OptiPlex-5070:-/Documents/1BM21CS083S
```

MiscecsegoMiscecse-Optiblex-50/0:-/Documents/1BM21C50835 lex anbn.l

Design a suitable grammar for evaluation of arithmetic expression having +, -, x, /, 1, a operator. A highest precedence and right . Accord highest and left \*, I second highest and left +, - lowest precedente and left prool. L Loption noyyurap Hindude "y. tab. h" prof("In Industry request 1.3 [0-9] + {yylval = atoi (yytext); return NUM; } [it]; In return o. . return yytext[0]. prol.y 4.8 #holude < stdio h> y token NUM 1+11-1 7. Wft (x) (/ xleft 1. left 111 1/night 1.4. expre: e { prints ("valid expression (n"); prints ("Result :xd (n", s) return 0,3 e: e'+' e 9\$ = 11 + \$3;3 l e '- ' e { p\$ = \$ 1 - \$3;} 1 e x'e { \$\$ = \$1 × \$3; } le'/'e 1 \$\$ 2 \$1/\$3;3 le' 1.1 e [\$\$ 2 \$1 %. \$3) y

\$1e'n'e } \$\$ = \$3 ^\$1;3 INUM \$\$\$ 2\$1;3 int main() printf("Inthter an axithmetic expression (no), gypanse (1) xeturn 0; int yyerror() print("In Invalid repression in"). 3 return 0, 0-974 Leglard = adoi (4/fet) : solum that an arithmetic expression 546 Valid expression Recut: 11 the an withmetic expression 5+6+\* Invalid expression

of flegill and commission of the flegill to me

14 6 4 32 5 31 + 123

```
bmscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:-/Documents$ lex proo1.1
bmscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/Documents$ vacc -d proo1.v
bmscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:-/Documents$ gcc lex.yy.c y.tab.c
v.tab.c: In function 'vvparse':
y.tab.c:1022:16: warning: implicit declaration of function 'yylex' [-Wimplicit-function-declaration]
 1022
             vvchar = vvlex ():
v.tab.c:1205:7: warning: implicit declaration of function 'vverror'; did you mean 'vverrok'? [-Wimplicit-function-declaration]
            yyerror (YY ("syntax error"));
 1205
bmscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:-/Documents$ ./a.out
Enter an arithmetic expression
5+6
Valid expression
Result: 11
bmscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/Documents$ ./a.out
Enter an arithmetic expression
5*6-2
Valid expression
Result: 28
bmscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:-/Documents$ ./a.out
Enter an arithmetic expression
5-6+*
Invalid expression
bmscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:-/DocumentsS
```

```
Write a gace program to generate syntax tree for the given
outhoretic expression
 7.5
  #include "y.tab. h"
  extern int yylval;
   4. 4
            Typial = atoi (gytext); return digit; }
   [0-9]4
   [Ity]
    [in] neturo,
     . return yetert [0].
    int yyurapi)
  Poy
  1.8
   Hindude ( math.h>
   #induderchipe. h>
  #holude < stdio. h >
   Hindudis stalib. 4>
   #indude cstring. h>
   whent tree-noch
        char val(10);
                         register testind by val)
        int lc;
                                 syn tree (ind) . In
        int re;
     int ind:
     struct breenede syn tree [100];
      void buf print tree (int curs ind);
           mknode (int le, int re, char val (10));
       4. 4
     x token digit
```

```
S:E fmy print true ($1); y
 E: E'+'T {$$ = mknode ($!, $3, "+");;3
  17 99$ = $1; 3
 T: T'*' F $$$ = mknode ($1,$3,"*");; 3
  IF 9$$ , $1,3
 E: ,(,E,), &$$ = $5:7
 Idigit schoo buffer (10); sprints (by, 1.d", gylval),
                    $ $ = mknode (-1, -1, buf); }
 int main!)
    ind=0.
     print (" Enter an expression In");
     gypanse ();
in yyerror ()
 print (" NITH Emy In").
 int anknode (int le, int re, chave valleon)
    stropy (syn-true lind). val, val);
    syntae Find). le. = le;
    Syntre [ind ). oc = oc.
     ind+; return in-1;
roid my-print-too (int cur-ind)
( if law_ind == -1)
      retur ;
   il (syn_ tree (cur_ind). le ==-1 60 syn_ bee (cur_ind).re=
      print l'Digit Node -> Index : Yid; Value : XSIN;
              curind, syntree (cur. ind). val);
```

print ("Operator Node -> Index: Y.d, Value i/3, left child Index: vd, Right child Index: v.d In" (wor ind, synthe (cur\_ind) val, synthe [cur\_ind) 10, syn tru (cur ind) rc); my-print-tree (syn-tree (con-ind). 10) my-print tree (syntree (curind I.re) Output: Enter on expression 4+6+9 operate Node -> Index: 4, Value: +, left shill Index: 9,

Right child Index: 3

Digital Node -> Index: 0, Value: 4 Operate Node -> Index: 3, Value: \*, left child Index: 1, Right Child Index: 2 Digit Node → Index: 1, Value: 6
Digit Node → Index: 2, Value: 9

give besuch " Honey I'm 80 AAAAA: Hore

```
bmscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/Documents ./a.out
Enter an expression
4+6*9
Operator Node -> Index : 4. Value : +. Left Child Index : 0.Right Child Index : 3
Digit Node -> Index : 0, Value : 4
Operator Node -> Index : 3. Value : *. Left Child Index : 1.Right Child Index : 2
Digit Node -> Index : 1. Value : 6
Digit Node -> Index : 2. Value : 9
bmscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:-/Documents
```

```
Use YACC to convert: Infx expression to Paifix
expression
  infix to postfix. 1
  # Andude < stdin. h>
  Hindudo (stalib. h)
  Hindred "y tab. h"
  exten int youal;
   Y. Y.
  [0-9] + Syyval = atoi (yytext); xetur num;)
  [It ]:
  in franction 0; 3
   · of notion gytex + [0]; 5
       yyarapi)
  infix- to-prefix.y
   1. 8
   #indude soldin.h>
 #include Coldlib.h>
  int gyperor (const chan 45).
        gylex (void)
   1,5
   1. token num
 1. left
         1 * 1 / 1
 1. left
  1. left
  1. left
          -1)
 1, right
  1.1.
```

```
* print ("n"); ]
    e: e '+ '+ Sprint ("+");3
      e'-'t & print ("-"):3
     t: + * 1 { print(" * ") ; 3
     1+ 1/h & pring (1/1); 3
     14
     h: f inh sprint ("1"); }
    f: ('e')'
(num Sprint) ("Y.d" $1):3
    void mais ()
     prints ("Enter an infox expression: 10").
    194 gyerror (cont char is)
       print (" Invalid infix expression . (In").
    3 returns,
Output
Enter an infix expression:
2+4 *5
245*+
Enter an infix expression:
3+6*2-1/3
362*+13/-
```

```
bmscecse@bmscecse-OptiPlex-5070:-/Documents/18M21CS8835 lex inflx to bostflx.t
bmscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21CS083$ yacc -d infix to postfix
bmscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21CS083$ gcc lex.yy.c y.tab.c
bmscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21CS083$ ./a.out
Enter an infix expression:
2+4*5
245*+
bmscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21C5083$ ./a.out
Enter an infix expression:
3+6*2-1/3
362*+13/-
```

3-Address codo for a given Use 4ACC to generate expression. 3 add wode, l #include Kotdio.h> #include < Adlib. ho (1 to Moint 3) # ndude "y. + ab. h 4 (0) 1) Horal 3 exten in yylval; extern char idensis), d (0-9]+ ( ( ( ) Hough d' ? } a [Q-ZA-Z]+ fd3 fyy wal atoi (gytext); return digit; 5 fal & stropy (iden, gylext); gyval=1; return id; [4] 5;3 In return o return eyext (0). in yourapl) xetur 1. 3 add code, y Hindlede smath, hs Hindude < chyp. ho a Hindlede Soldio, ho Also 184 50 ich yyenor (chan 45). int yylex (void); int. van-ent=0; chas identical. 1.3

```
id
xtoken digit
sid '= 'E spring ("x.s= +x.d \n" iden, van.crt - D; 5
E) E 1+1 T S$$ = van-cont; van-cont ++; probable ("1+ x de
1B'-' T & $$ $= var_cot; var_cot +1; printh ("+1/d =
              +xd -(xd)10", $$, $1, $3);3
11 2 44 > 41:7
 T: T '* 1 F f $ = van-crd; van-crd ++; printly ("+1.d=+1.d+
                     + 4. d; In", $$, $1,$3); 5
 IT 1/ F f $$ > var-cnt, varient ++; printf ("+ 1/d=+1/d)
                    tyd; \n" ps, $1,$3).5
       fp$ > $1;4
  F: P'n' F 9$$ = van-ent; van-ent ++; printb(" + vd =
                    txd ^txd >\n', $$, $1,$3);3
  1 P f$$,$1;5
  P. '(' E')' f$$ 2$2,53
  I digit 5$4 > van-ent; van-ent ++. print (41.0=1.0)
                                            かかりうう
  int main!)
 { var-crt =0;
     prohity ("Enter on expression: 10");
      Jypanes;
      setuen o.
 int yyerror (char *s)
    print ("Invalid expression!").
    return 0,
```

1. token

Output Enter on expression: az 8+9-2 to = 8, t129, £2 = to+t1; £3 20; t4 > 12-t3; a> 14 spity And in german I german ette I w I Pix + pixt, buil in to rear i prove 6 \$ 5 gill ( + 19 - 19 ). poly ("Ester or experience No").

```
mscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21CS0B3S lex 3addcode.l
mscecse@bmscecse-OptiPlex-5070:-/Documents/18M21C5083$ vacc -d 3addcode.v
mscecse@bmscecse-OptiPlex-5070:-/Documents/18M21CS083$ gcc lex.yy.c y.tab.c
mscecse@bmscecse-OptiPlex-5070:-/Documents/1BM21CS083$ ./a.out
nter an expression:
=8+9-2
0 = 8:
1 = 9:
2 = t0 + t1:
3 = 2:
4 = t2 - t3;
=t4
mscecse@bmscecse-OptiPlex-5070:~/Documents/1BM21CS083$ ./a.out
nter an expression:
=2^3/23+5
9 = 2:
1 = 3:
2 = t0 ^ t1:
3 = 23:
= t2 / t3:
5 = 5:
5 = t4 + t5;
:t6
```