```
Title :
                                                                       Date:
                                                                       Page No: 2
                    -while (tempso] = 'J')
                       str[j+t] = temp[s+t];
                     sto[j] = 1\0';
                     printf(" '/d \t \t '/s \t\t indea\n', 1, sh);
                z
                 else
                 £
for ( k=0; k< #; k++)
                         if ( |strcmp(c(k7.str))
                           bacak;
                     9F(k==8)
                        printf(" 1/d/t/t 1/s/t/t identifier/n", 1, sh);
                     ebe
                        printf("/d \t \t '/s \t \t keyword \n", 1, str);
             elsesf ( isdigit (tempsis]))
int 15-float =0, 15-exp=0;
                while ( is digit ( temp[5]))
                    517[j++] =temp[5++7)
                 if (temp[i] == (.))
                     15-float=1)
                     Sto [j++] = temp[s++];
                     while ( ssdigit (temp[s]))
                       Str[j++] = temp[5++];
```

```
Date:
         Title:
                                                                       Page No: 4
                  printf (" 1/d \t\t1/\t\t end of comment\t\n",1);
               else sf ((temp[s] == '/') & (temp[s+1] == '/'))
                   printf("/d/1/t/\t\single line comment \t\n",();
                   bocaki
               else sf( temp[s] == "")
or Fig
               ٤
                   j=0;
                    printf (" "/d/t/t/"/t/t start of tsteral/t/n",1);
                   9++5
                    white ( temp [1] | = "", & temp [1] | = (\n')
                       sta [ ]++] = temp[ 5 + 7] 5
                   printf("/,d\t\t'/s\t\t end of string literal \n", 1, sto);
                    9++;
               else
               Ł
                  j=0;
                   stolj++] = temp[s];
                   switch (temp[1])
                      case '+':
                       cose (-) :
                      case (A):
                       case '/':
                       case '=';
                       case '>':
                       case '<':
                      case '%': { if ((temp[s] == 1 > 1 | temp[s] == 1 < 1 | temp[s] == 1 = 1) 92
                                      temp[[+1] == (=')
```

```
Title:
                                                              Page No: 8
     str[]++]= temp[++7;
    5ta[]]='\0';
   if ( temp[i] == '(')
       printf ( "-/-d \t\t :/-s \t\t array\n",1,sto);
       j=0
       while ( temp[i] ] = ']')
          56[j++] = temp[s++7;
       5 to [j] = '\o';
      printf("/bd/t/t //s/t/t index/n",1,5ti)s
  z
 else
    for (K = 0; K < 38; K++)
        If (!strcmp(c(k],str))
        breaks
     if( K ==38)
        printf( "1/d/t/t 1/s/t/t identifies In", 1, str);
       printf("1.dlt \t: 1.5xtlt keyword \n", 1, sto);
 ŀ
else if ( isdigit (temp[i]))
   int 15-float=0, 55-exp=0;
   while ( isdigit ( temp [s]))
      sh [++] = temp[ ++ ]>
    z
```

```
Title:
                                                            Page No: 10
  Ę
     9++5
     if( temp[i] == (10))
        fgets (temp, 100, fp);
         14+5
         1=0;
 printf("1.d \t\t"/\t\t end of comment\t\n",();
else if ((templis)==1/1) 42 (temp[5+1]==1/1))
   printf(""/d/t/t ///t/t single line comment/t/n",1);
   breaks;
else sf( temp[s7 =='a')
    j=0;
    pointf(" 1, d ) t\t \" \t\t start of string literal \t\n", ();
    while (kmp[s]!= "" 9% temp[s]! = "(n))
        str [ ] ++ ] = temp[ 5++ ];
    Str[j]='\0';
    printf( "1.d \t \t 1.5 \t + \t end of string literal \n ",1,5to );
    9445
else
  j=0;
  Str[j++ ] = temp[s])
  switch ( temps 1)
```

```
Title :
                                                           Date :
                                                           Page No: 12
     3
  fclose (fp);
z
txt · tuqui
int main()
   hi welcome
   */
  // compiler design
  printf("co");
   int a, b;
   float c = 10.6E+7;
Output:
                                Lezeme
                Token
Line No
                                Keyword
                int
                                Keyword
                main
                                 parenthesis
                 C
                                 posenthesis
                 (
                                 parenmesis
                 ٤
                                 stort of comment
                                 end of comment
                4/
                                 single line of comment
                //
                                 keyword
               pannif
                                 parenthesis
                (
                                 start of string literal
                4
                                  end of string literal
               (D)
                                  paventhesis
```

7	3	punct symbol
ε	tot	keyword
8	а	sdentilier
	,	special symbol
8	b	identifier
8	-	punct symbol
8	;	idephilier
9	float	identifier
9	c	
9	=	operator
a	10.66+7	exponential
•	3	punct symbol
9	_	pasenthesis
10	ን	r

73

```
Title:
                                                          Date:
                                                          Page No: 15
         { printf(" lineno = 1/d \t 1/5 \t\t assignment oper \n", kneno, yytert);}
 "+" | "-" | " /" { printf(" lineno = 1.d \t 1.5 \t\t operator \n", lineno,
                       yyeat) }
 "["|"]"|"{"|"3"|"(")")" {printf("Ineno="/-d \t\t -/-s\t parenthesis\n",
                                  kneno, yytext);}
       { printf ("(ineno= 1 d /t/t 7.5 /t/t keyword \n", (ineno, yytext); }
{kw}
{any {printf("Isneno= '/d \t'/s \t\t array\n", Isneno, yytext);}
fst & { printf ( " Isneno = 1/d \t 1/.5 \t\t string (steral \n", lineno, yytext); }
        { printf ( " Isneno = 7-d \t 7.5 \t\t identifier \n", Isneno, yyteat); }
 4563
four } { printf(" kineno= 1.d \t 1.5 \t \t number \n", kineno, yytext); }
1....
main (int argo, char **argv)
   if (orge >1)
       yysn = fopen ( argv [1], "3");
    else
       yyin = stdin;
   yyle= ();
yywrap ()
  exit(0);
input tat
int main()
   hi welcome
   11 compiler Design
   printf("CD");
```

- - BI-201-3

1

```
Title: Recursive Decent Parser
                                                                  Date: 5/9/24
                                                                  Page No: /9
  Develop a Recursive Decent Ponson for statements in C.
  * include <stdio.h>
  & include conio h>
  void E();
  void Eicis
  void TC) 5
  void TI();
  void FU;
 void match (char);
for flag =1;
 char lits
 main()
   printf("Enter input string\n");
scanf("1-c", y1);
   E();
 3
void match (chas t)
1 x+(0==t)
   scanf( "1.c", 80)5
    else
   £ +1ag = 0;
   3
z
void E()
£ T();
  EI();
  if ( (1=='$') & (flag!=0))
  printf( " successful \n ");
  print ( "unsuccessful (n");
```

```
Grammar!
E -> TE1
\epsilon' \longrightarrow + \tau \epsilon' / \epsilon
T \longrightarrow FT'
T' \longrightarrow *FT'/\epsilon
 F -> i
F \rightarrow (e)
                                                         FOLLOW
                               First
                                                       を 歩, ) 子
                             €1,63
E - TE'
                                                        ٤ $,73
                             を十、巨子
 E' -> + TE'/E
                                                       { +,$,73
 T -> FT'
                             もかもろ
                                                        を 十, ま, ) 多
 \tau' \longrightarrow *FT'/\epsilon
                             € *, € 4
                                                        を *,+,あ,)3
 F \rightarrow i
                             £ 5 3
                                                        をナッキ、まりうる
  F → (E)
                             € (3
            80
                                         (
                                                     >
                                        € → re¹
          E-> re1
                                                   e'->E
   €1
                   E' -> +TE'
         7 → FT 1
  TI
                   T' > E T > * FT'
         F->1
                                       F → (E)
```