

```

% {
    int newLine = 0;

    int space = 0;

    int c = 0;

    int tabs = 0;

    int words = 0;

% }

%%

[ ] { space++;}

[\n] { newLine++;}

[\t] { tabs++;}

[^\n\t ]+ { words++;c += yyleng;}

%%

int yywrap() {return 1;}

int main(){

    printf("Enter the input: ");

    yylex();

    printf("No. of chars: %d\n",c);

    printf("No. of spaces: %d\n",space);

    printf("No. of new lines: %d\n",newLine);

    printf("No. of words: %d\n",words);

    printf("No. of tabs: %d\n",tabs);

}

```

## OUTPUT

```
[kartick@kartick-11118 TermWork]$ lex ques1.1
[kartick@kartick-11118 TermWork]$ gcc lex.yy.c
[kartick@kartick-11118 TermWork]$ ./a.out
Enter the input: Hello how          are          you
I am      doing fine
No. of chars: 26
No. of spaces: 8
No. of new lines: 2
No. of words: 8
No. of tabs: 3
```

```

% {

    #include<stdio.h>

    #include<string.h>

% }

%%

"/*"^[^*]*"/";

"/"^[^\n]*  ;

. {

    fprintf(yyout,"%s",yytext);

}

%%

int yywrap() { return 1; }

int main() {

    extern FILE *yyin, *yyout;

    yyin = fopen("input.c", "r");

    yyout = fopen("out.c", "w");

    yylex();

    fclose(yyin);

    fclose(yyout);

    return 0;

}

```

## INPUT

```
#include<stdio.h>
int main(){
    /*Hello world*/
    //Hello World
    printf("Hello World");
    return 0;
}
```

## OUTPUT

```
#include<stdio.h>
int main(){

    printf("Hello World");
    return 0;
}
```

```
% {  
    #include<stdio.h>  
    #include<stdlib.h>  
% }  
%%  
"<"[^>]*">" {printf("%s\n",yytext);}  
.  
%;  
%%  
int yywrap(){  
    return 1;  
}  
int main(){  
    extern FILE *yyin;  
    yyin = fopen("htmlInput.html","r");  
    yylex();  
    fclose(yyin);  
}
```

## INPUT

```
<html>
  <head>
    <title>Document</title>
  </head>
  <body>
    <div>Hello How are you</div>
  </body>
</html>
```

## OUTPUT

```
[kartick@kartick-11118 TermWork]$ lex ques4.1
[kartick@kartick-11118 TermWork]$ gcc lex.yy.c
[kartick@kartick-11118 TermWork]$ ./a.out
<html>

<head>

<title>
</title>

</head>

<body>

<div>
</div>

</body>

</html>
```

```
% {  
    #include<stdio.h>  
% }  
%%  
[0-9]+ {printf("%s is an Integer\n",yytext);}  
[0-9]+". "[0-9]* {printf("%s is Float\n",yytext);}  
%%  
int yywrap(){return 1;}  
int main(){  
    printf("Enter the input: ");  
    yylex();  
}
```

## OUTPUT

```
[kartick@kartick-11118 TermWork]$ lex ques5.1
[kartick@kartick-11118 TermWork]$ gcc lex.yy.c
[kartick@kartick-11118 TermWork]$ ./a.out
Enter the input: 123
123 is an Integer

123.45
123.45 is Float

0.456
0.456 is Float

123321
123321 is an Integer
```



```
% {  
    #include<stdio.h>  
% }  
  
%%  
  
[a-zA-Z][a-zA-Z0-9]* {printf("%s is an identifier\n",yytext);}  
  
.* {printf("%s is not an identifier\n",yytext);}  
  
%%  
  
int yywrap(){return 1;}  
  
int main(){  
    printf("Enter identifier\n");  
    yylex();  
}
```

## OUTPUT

```
[kartick@kartick-11118 TermWork]$ lex ques6.1
[kartick@kartick-11118 TermWork]$ gcc lex.yy.c
[kartick@kartick-11118 TermWork]$ ./a.out
Enter identifier
abc
abc is an identifier

1avc
1avc is not an identifier

abc12
abc12 is an identifier

Aab12
Aab12 is an identifier

aAbc12
aAbc12 is an identifier
```

```

% {
    #include<stdio.h>

% }

%%

[0-9]+ {printf("%s is an Integer\n",yytext);}
[0-9]*"."[0-9]+ {printf("%s is a Float\n",yytext);}
"if"|"else"|"for"|"while"|"int"|"float" {printf("%s is a Keyword\n",yytext);}
"+"|"-"|"*"|" "/"|"="|"<"|">"|"++"|"--" {printf("%s is an Operator\n",yytext);}
", "|" ";" {printf("%s is A Seperator\n",yytext);}
[a-zA-Z][a-zA-Z0-9]* {printf("%s is an Identifier\n",yytext);}
[ ] {printf("");}
. {printf("Invalid Input\n");}

%%

int yywrap(){return 1;}

int main(){
    printf("Enter the input: \n");
    yylex();
}

```

## OUTPUT

```
[kartick@kartick-11118 TermWork]$ lex ques7.1
[kartick@kartick-11118 TermWork]$ gcc lex.yy.c
[kartick@kartick-11118 TermWork]$ ./a.out
Enter the input:
int a,b,c;
int is a Keyword
a is an Identifier
, is A Seperator
b is an Identifier
, is A Seperator
c is an Identifier
; is A Seperator
```

```

% {

    #include <stdio.h>

    #include <stdlib.h>

    #include <string.h>

    int newLine = 0;

    int space = 0;

    int c = 0;

    int words = 0;

% }

%%

[ ] { space++;}

[\n] { newLine++;}

[^\n\t ]+ { words++;c += yyleng;}

%%

int yywrap() {

    return 1;

}

int main(){

    extern FILE *yyin,*yyout;

    yyin=fopen("input.txt","r");


    yyout=fopen("output.txt","w");

    yylex();

```

```
fprintf(yyout,"No. of chars: %d\n",c);  
fprintf(yyout,"No. of spaces: %d\n",space);  
fprintf(yyout,"No. of new lines: %d\n",newLine);  
fprintf(yyout,"No. of words: %d\n",words);  
fclose(yyin);  
fclose(yyout);  
return 0;  
}
```

## INPUT

 input.txt

```
1  Hellow      how are   you
2  I am      doing fine
```

## OUTPUT

 output.txt

```
1  No. of chars: 27
2  No. of spaces: 15
3  No. of new lines: 2
4  No. of words: 8
5  
```

```

% {
    #include<stdio.h>

% }

%%

[ ]+ {
    fprintf(yyout," ");
}

. {
    fprintf(yyout,"%s",yytext);
}

%%


int yywrap(){
    return 1;
}

int main(){
    extern FILE *yyin,*yyout;
    yyin = fopen("input.txt","r");
    yyout = fopen("output.txt","w");
    yylex();
    fclose(yyin);
    fclose(yyout);
}

```




## INPUT

 input.txt

```
1  Hellow      how are   you
2  I am      doing fine
```

## OUTPUT

 output.txt

```
1  Hellow how are you
2  I am doing fine
```

```

% {
    #include<stdio.h>

    #include<string.h>

% }

%%

"/*"^[^*]*"/"  ;

"//[^\n]*"      ;

.      { fprintf(yyout,"%s",yytext); }

%%

int yywrap() { return 1; }

int main() {
    extern FILE *yyin, *yyout;

    yyin = stdin;

    yyout = fopen("out.c", "w");

    yylex();

    fclose(yyin);

    fclose(yyout);

    return 0;

}

```

## INPUT

```
• [kartick@kartick-11118 TermWork]$ lex ques10.1
• [kartick@kartick-11118 TermWork]$ gcc lex.yy.c
• [kartick@kartick-11118 TermWork]$ ./a.out
#include<stdio.h>
int main(){
    /*Hello world*/
    //Hello World
    printf("Hello World");
    return 0;
}
```

## OUTPUT

```
C out.c > ...
1  #include<stdio.h>
2  int main(){
3
4
5      printf("Hello World");
6      return 0;
7  }
```

```

% {
    #include<stdio.h>

    #include<stdlib.h>

% }

% %

"<[^>]*">"| " {fprintf(yyout,"%s",yytext);}

. ;

% %

int yywrap(){
    return 1;
}

int main(){
    extern FILE *yyin,*yyout;
    yyin = fopen("htmlFile.html","r");
    yyout = fopen("htmlTags.txt","w");
    yylex();
    fclose(yyin);
    fclose(yyout);
}

```

## INPUT

```
<html>
  <head>
    <title>Document</title>
  </head>
  <body>
    <div>Hello How are you</div>
  </body>
</html>
```

## OUTPUT

```
htmlTags.txt
1  <html>
2    <head>
3      <title></title>
4    </head>
5    <body>
6      <div>    </div>
7    </body>
8  </html>
```

```

% {
    #include<stdio.h>

    #include<stdlib.h>

% }

%%

[0-9]+ {fprintf(yyout,"%s is an Integer\n",yytext);}

[0-9]*"."[0-9]+ {fprintf(yyout,"%s is a Float\n",yytext);}

"if"|"else"|"for"|"while"|"int"|"float" {fprintf(yyout,"%s is a Keyword\n",yytext);}

"+"|"-"|"*"|" "/"|"=="|"="|"<"|">"|"++"|"--" {fprintf(yyout,"%s is an Operator\n",yytext);}

", "|" ";" {fprintf(yyout,"%s is A Seperator\n",yytext);}

[a-zA-Z][a-zA-Z0-9]* {fprintf(yyout,"%s is an Identifier\n",yytext);}

[ ] {printf("");}

. ;

%%

int yywrap(){return 1;}

int main(){
    extern FILE *yyin,*yyout;

    yyin = fopen("token.c","r");
    yyout = fopen("tokens.txt","w");

    yylex();

    fclose(yyin);

    fclose(yyout);

}


```

## INPUT

**C** token.c > ...

```
1  int p=1,d=0,r=4;
2  float m=0.0, n=200.0;
3  while (p <= 3)
4  { if(d==0)
5    { m= m+n*r+4.5; d++; }
6    else
7    { r++; m=m+r+1000.0; }
8    p++; }
```

## OUTPUT

 tokens.txt

```
1  int is a Keyword
2  p is an Identifier
3  = is an Operator
4  1 is an Integer
5  , is A Seperator
6  d is an Identifier
7  = is an Operator
8  0 is an Integer
9  , is A Seperator
10 r is an Identifier
11 = is an Operator
12 4 is an Integer
13 ; is A Seperator
14
15 float is a Keyword
16 m is an Identifier
17 = is an Operator
18 0.0 is a Float
19 , is A Seperator
20 n is an Identifier
21 = is an Operator
22 200.0 is a Float
23 ; is A Seperator
24
25 while is a Keyword
26 p is an Identifier
27 < is an Operator
28 = is an Operator
29 3 is an Integer
```



```

% {
    #include<stdio.h>

    int words = 0;
% }

%%

[^\t\n ]+ {
    int c = yyleng;
    if(c == 5){
        words++;
    }
}

%%

int yywrap(){
    return 1;
}

int main(){
    printf("Enter the input: \n");
    yylex();
    printf("No. of words having lenght 5-> %d\n",words);
}

```

## OUTPUT

```
● [kartick@kartick-11118 TermWork]$ lex ques13.1
● [kartick@kartick-11118 TermWork]$ gcc lex.yy.c
● [kartick@kartick-11118 TermWork]$ ./a.out
Enter the input:
Hello world how are you doing I am also fine

No. of words having lenght 5-> 3
```

```

% {
    #include<stdio.h>

    int v = 0;

    int c = 0;
% }

% %

[aeiouAEIOU] {
    v++;

    printf("Vowel-> %s\n",yytext);
}

[^ aeiouAEIOU] {
    c++;

    printf("Consonant %s\n",yytext);
}

% %

int yywrap(){
    return 1;
}

int main(){
    printf("Enter the input: \n");

    yylex();

    printf("Total Vowels -> %d\n",v);

    printf("Total Consonants -> %d\n",c);
}

```

## OUTPUT

```
● [kartick@kartick-11118 TermWork]$ lex ques14.1
● [kartick@kartick-11118 TermWork]$ gcc lex.yy.c
○ [kartick@kartick-11118 TermWork]$ ./a.out
Enter the input:
Hello world how are you
Consonant H
Vowel-> e
Consonant l
Consonant l
Vowel-> o
  Consonant w
Vowel-> o
Consonant r
Consonant l
Consonant d
  Consonant h
Vowel-> o
Consonant w
  Vowel-> a
Consonant r
Vowel-> e
  Consonant y
Vowel-> o
Vowel-> u
```

```

% {

    #include<stdio.h>

    #include<string.h>

% }

%%

[0-9]* {

    int n = yyleng;

    int flag = 1;

    for(int i = 0;i <= n/2;i++){

        if(yytext[i] != yytext[n-i-1]){

            flag = 0;

            break;

        }

    }

    if(flag){

        printf("%s is a Pallindrome\n",yytext);

    }

    else{

        printf("%s is not a Pallindrome\n",yytext);

    }

}

. {

    printf("Please enter number\n");

}

%%

```

```
int yywrap(){  
    return 1;  
}  
int main(){  
    printf("Enter the Input: \n");  
    yylex();  
}
```

## OUTPUT

```
● [kartick@kartick-11118 TermWork]$ lex ques15.1
● [kartick@kartick-11118 TermWork]$ ./a.out
Enter the Input:
12321
12321 is a Pallindrome

123123
123123 is not a Pallindrome

121
121 is a Pallindrome
```

```

% {
    #include<stdio.h>

    #include<string.h>
% }

%%

[a-zA-Z0-9 ]* {
    int n = yyleng;
    int flag = 1;
    for(int i = 0;i <= n/2;i++){
        if(yytext[i] != yytext[n-i-1]){
            flag = 0;
            break;
        }
    }
    if(flag)
        printf("%s is a Pallindrome\n",yytext);
    else
        printf("%s is not a Pallindrome\n",yytext);
}

%%

int yywrap(){
    return 1;
}

int main(){
    printf("Enter the Input: \n");
    yylex();
}

```



## OUTPUT

```
• [kartick@kartick-11118 TermWork]$ lex ques16.1
• [kartick@kartick-11118 TermWork]$ gcc lex.yy.c
• [kartick@kartick-11118 TermWork]$ ./a.out
Enter the Input:
ABCAB
ABCAB is not a Pallindrome

ABCBA
ABCBA is a Pallindrome

ABCABC
ABCABC is not a Pallindrome
```

```

% {
    #include<stdio.h>

% }

%%

[^@]+"@"["gmail"|"hotmail"]+"."["com"|"org"]+ {
    printf("%s is a valid email\n",yytext);
}

[^\\n]+ {
    printf("%s is an Invalid Email\n",yytext);
}

%%

int yywrap(){
    return 1;
}

int main(){
    printf("Enter the Input: \n");
    yylex();
}

```

## OUTPUT

- [kartick@kartick-11118 TermWork]\$ lex ques17.1
- [kartick@kartick-11118 TermWork]\$ gcc lex.yy.c
- [kartick@kartick-11118 TermWork]\$ ./a.out

Enter the Input:

abc123@gmail.com

abc123@gmail.com is a valid email

abc12@xyz.co

abc12@xyz.co is an Invalid Email

```

% {
    #include<stdio.h>

    #include<string.h>

    int minLen = 1e9;

    char word[100];
% }

%%

[^\t\n]* {
    if(yyleng < minLen){
        minLen = yyleng;
        strcpy(word,yytext);
    }
}

%%

int yywrap(){
    return 1;
}

int main(){
    printf("Enter the Input: \n");

    yylex();

    printf("Shortest Word-> %s\n",word);
}

```

## OUTPUT

```
• [kartick@kartick-11118 TermWork]$ lex ques18.1
• [kartick@kartick-11118 TermWork]$ gcc lex.yy.c
• [kartick@kartick-11118 TermWork]$ ./a.out
Enter the Input:
Hello world how are you do

Shortest Word-> do
```

```

% {

    #include<stdio.h>

    #include<string.h>

% }

%%

[a-zA-Z0-9 ]* {

    int n = yyleng;

    int org = 0;

    for(int i = 0;i < n;i++){

        org = org*10 + (yytext[i]-'0');

    }

    int temp = org*org;

    int check = 0;

    while(temp){

        int rem = temp%10;

        check += rem;

        temp /= 10;

    }

    if(check == org){

        printf("%s is a Neon Number\n",yytext);

    }

    else{

        printf("%s is not a Neon Number\n",yytext);

    }

}

%%

```

```
int yywrap(){  
    return 1;  
}  
int main(){  
    printf("Enter the Input: \n");  
    yylex();  
}
```

## OUTPUT

```
• [kartick@kartick-11118 TermWork]$ lex ques19.1
• [kartick@kartick-11118 TermWork]$ gcc lex.yy.c
• [kartick@kartick-11118 TermWork]$ ./a.out
Enter the Input:
9
9 is a Neon Number

12
12 is not a Neon Number
```



```

% {
    #include<stdio.h>

    #include<string.h>
% }

%%

[a-zA-Z0-9 ]* {
    int n = yyleng;
    int org = 0;
    for(int i = 0;i < n;i++){
        org = org*10 + (yytext[i]-'0');
    }
    int temp = org*org;
    if(org%10 == temp%10){
        printf("%s is an Automorphic Number\n",yytext);
    }
    else{
        printf("%s is not an Automorphic Number\n",yytext);
    }
}

%%

int yywrap(){
    return 1;
}

int main(){
    printf("Enter the Input: \n");
    yylex();
}

```

## OUTPUT

- [kartick@kartick-11118 TermWork]\$ lex ques20.1
- [kartick@kartick-11118 TermWork]\$ gcc lex.yy.c
- [kartick@kartick-11118 TermWork]\$ ./a.out

Enter the Input:

11

11 is an Automorphic Number

12

12 is not an Automorphic Number

17

17 is not an Automorphic Number

16

16 is an Automorphic Number

```

% {
    #include<stdio.h>

% }

%s A B C

%%

<INITIAL>0 BEGIN B;
<INITIAL>1 BEGIN A;
<INITIAL>\n BEGIN INITIAL; {printf("Accepted\n");}
<A>0 BEGIN C;
<A>1 BEGIN INITIAL;
<A>\n BEGIN INITIAL; {printf("Not Accepted\n");}
<B>1 BEGIN C;
<B>0 BEGIN INITIAL;
<B>\n BEGIN INITIAL; {printf("Not Accepted\n");}
<C>0 BEGIN A;
<C>1 BEGIN B;
<C>\n BEGIN INITIAL; {printf("Not Accepted\n");}

%%

int yywrap(){return 1;}

int main(){
    printf("Enter the input: ");
    yylex();
}

```

## OUTPUT

```
• [kartick@kartick-11118 TermWork]$ lex ques21.1
• [kartick@kartick-11118 TermWork]$ gcc lex.yy.c
• [kartick@kartick-11118 TermWork]$ ./a.out
Enter the input: 1100
Accepted
101
Not Accepted
1001
Accepted
100010001001
Accepted
10000001
Accepted
1000001
Not Accepted
```

```
% {  
    #include<stdio.h>  
% }  
  
%s A B C D E F G  
  
%%  
  
<INITIAL>0 BEGIN A;  
<INITIAL>1 BEGIN D;  
<INITIAL>\n BEGIN INITIAL; {printf("Not Accepted\n");}  
<A>0 BEGIN D;  
<A>1 BEGIN B;  
<A>\n BEGIN INITIAL; {printf("Not Accepted\n");}  
<B>1 BEGIN C;  
<B>0 BEGIN D;  
<B>\n BEGIN INITIAL; {printf("Not Accepted\n");}  
<C>0 BEGIN E;  
<C>1 BEGIN C;  
<C>\n BEGIN INITIAL; {printf("Not Accepted\n");}  
<D>0 BEGIN D;  
<D>1 BEGIN D;  
<D>\n BEGIN INITIAL; {printf("Not Accepted\n");}  
<E>0 BEGIN F;  
<E>1 BEGIN G;  
<E>\n BEGIN INITIAL; {printf("Accepted\n");}  
<F>0 BEGIN F;  
<F>1 BEGIN G;  
<F>\n BEGIN INITIAL; {printf("Not Accepted\n");}
```

```
<G>0 BEGIN F;  
<G>1 BEGIN C;  
<G>\n BEGIN INITIAL; {printf("Not Accepted\n");}  
%%  
  
int yywrap(){return 1;}  
  
int main(){  
    printf("Enter the input: ");  
    yylex();  
}
```

## OUTPUT

```
• [kartick@kartick-11118 TermWork]$ lex ques22.1
• [kartick@kartick-11118 TermWork]$ gcc lex.yy.c
• [kartick@kartick-11118 TermWork]$ ./a.out
Enter the input: 0110
Accepted
011000110
Accepted
1101110
Not Accepted
011001
Not Accepted
011001100110
Accepted
```

```

% {
    #include<stdio.h>

% }

%s A B C

%%

<INITIAL>[a-zA-Z] BEGIN A;
<INITIAL>[0-9] BEGIN B;
<A>[a-zA-Z0-9] BEGIN A;
<A>\n BEGIN INITIAL; {printf("Identifier\n");}
<B>[0-9] BEGIN B;
<B>[.] BEGIN C;
<B>\n BEGIN INITIAL; {printf("Integer\n");}
<C>[0-9] BEGIN C;
<C>\n BEGIN INITIAL; {printf("Float\n");}

%%

int yywrap(){return 1;}

int main(){
    printf("Enter the input: ");
    yylex();
}

```



## OUTPUT

```
• [kartick@kartick-11118 TermWork]$ lex ques23.1
• [kartick@kartick-11118 TermWork]$ gcc lex.yy.c
• [kartick@kartick-11118 TermWork]$ ./a.out
Enter the input: 123
Integer
ABC
Identifier
ABC123
Identifier
123.45
Float
```

```

% {
    #include<stdio.h>

% }

% s A B C

% %

<INITIAL>a BEGIN B;
<INITIAL>b BEGIN A;
<INITIAL>\n BEGIN INITIAL; {printf("Accepted\n");}
<A>a BEGIN C;
<A>b BEGIN INITIAL;
<A>\n BEGIN INITIAL; {printf("Not Accepted\n");}
<B>b BEGIN C;
<B>a BEGIN INITIAL;
<B>\n BEGIN INITIAL; {printf("Not Accepted\n");}
<C>a BEGIN A;
<C>b BEGIN B;
<C>\n BEGIN INITIAL; {printf("Not Accepted\n");}

% %

int yywrap(){return 1;}

int main(){
    printf("Enter the input: ");
    yylex();
}

```

## OUTPUT

```
• [kartick@kartick-11118 TermWork]$ lex ques24.1
• [kartick@kartick-11118 TermWork]$ gcc lex.yy.c
• [kartick@kartick-11118 TermWork]$ ./a.out
Enter the input: aabb
Accepted
abb
Not Accepted
aaaabbabbabb
Accepted
abbbbabbba
Not Accepted
```

```
% {  
    #include<stdio.h>  
% }  
  
%s A B C D E F G  
  
%%  
  
<INITIAL>a BEGIN A;  
  
<INITIAL>b BEGIN INITIAL;  
  
<INITIAL>\n BEGIN INITIAL; {printf("Not Accepted\n");}  
  
<A>a BEGIN B;  
  
<A>b BEGIN C;  
  
<A>\n BEGIN INITIAL; {printf("Not Accepted\n");}  
  
<B>a BEGIN D;  
  
<B>b BEGIN E;  
  
<B>\n BEGIN INITIAL; {printf("Not Accepted\n");}  
  
<C>a BEGIN F;  
  
<C>b BEGIN G;  
  
<C>\n BEGIN INITIAL; {printf("Not Accepted\n");}  
  
<D>a BEGIN D;  
  
<D>b BEGIN E;  
  
<D>\n BEGIN INITIAL; {printf("Accepted\n");}  
  
<E>a BEGIN F;  
  
<E>b BEGIN G;  
  
<E>\n BEGIN INITIAL; {printf("Accepted\n");}  
  
<F>a BEGIN B;  
  
<F>b BEGIN C;  
  
<F>\n BEGIN INITIAL; {printf("Accepted\n");}
```

<G>a BEGIN A;

<G>b BEGIN INITIAL;

<G>\n BEGIN INITIAL; {printf("Accepted\n");}

%%

```
int yywrap(){
```

```
    return 1;
```

```
}
```

```
int main(){
```

```
    printf("Enter the Input\n");
```

```
    yylex();
```

```
    return 0;
```

```
}
```

## OUTPUT

```
• [kartick@kartick-11118 TermWork]$ lex ques28.1
• [kartick@kartick-11118 TermWork]$ gcc lex.yy.c
• [kartick@kartick-11118 TermWork]$ ./a.out
Enter the Input
abbbabb
Accepted
abbabbbb
Not Accepted
abbab
Not Accepted
abbaaa
Accepted
```

```

% {
    #include<stdio.h>

% }

%s A B C D

%%

<INITIAL>0 BEGIN A;
<INITIAL>1 BEGIN C;
<INITIAL>\n BEGIN INITIAL; {printf("Not Accepted\n");}
<A>0 BEGIN A;
<A>1 BEGIN B;
<A>\n BEGIN INITIAL; {printf("Not Accepted\n");}
<B>1 BEGIN B;
<B>0 BEGIN A;
<B>\n BEGIN INITIAL; {printf("Accepted\n");}
<C>0 BEGIN D;
<C>1 BEGIN C;
<C>\n BEGIN INITIAL; {printf("Not Accepted\n");}
<D>0 BEGIN D;
<D>1 BEGIN C;
<D>\n BEGIN INITIAL; {printf("Accepted\n");}

%%

int yywrap(){return 1;}

int main(){
    printf("Enter the input: ");
    yylex();
}

```

## OUTPUT

```
• [kartick@kartick-11118 TermWork]$ lex ques29.1
• [kartick@kartick-11118 TermWork]$ gcc lex.yy.c
• [kartick@kartick-11118 TermWork]$ ./a.out
Enter the input: 1000100101
Not Accepted
001000101
Accepted
0110010010
Not Accepted
111001000010
Accepted
```



```

% {
    #include<stdio.h>

% }

%s A B C D E

%%

<INITIAL>a BEGIN A;
<INITIAL>b BEGIN D;
<INITIAL>\n BEGIN INITIAL; {printf("Not Accepted\n");}
<A>a BEGIN D;
<A>b BEGIN B;
<A>\n BEGIN INITIAL; {printf("Not Accepted\n");}
<B>b BEGIN B;
<B>a BEGIN C;
<B>\n BEGIN INITIAL; {printf("Not Accepted\n");}
<C>a BEGIN E;
<C>b BEGIN B;
<C>\n BEGIN INITIAL; {printf("Accepted\n");}
<D>a BEGIN D;
<D>b BEGIN D;
<D>\n BEGIN INITIAL; {printf("Not Accepted\n");}
<E>a BEGIN E;
<E>b BEGIN B;
<E>\n BEGIN INITIAL; {printf("Not Accepted\n");}

%%

```

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

int main(){
    printf("Enter the input: ");
    yylex();
}
```

## OUTPUT

```
• [kartick@kartick-11118 TermWork]$ lex ques30.1
• [kartick@kartick-11118 TermWork]$ gcc lex.yy.c
• [kartick@kartick-11118 TermWork]$ ./a.out
Enter the input: aba
Accepted
abba
Accepted
abbbababbbaa
Not Accepted
abbabbababa
Accepted
```