

numlineswordscharacters.l

```
1  %{
2      #include<stdio.h>
3      int sc=0,wc=0,lc=0,cc=0;
4  %}
5
6  %%
7  [\n] { lc++; cc+=yyleng;}
8  [ \t] { sc++; cc+=yyleng;}
9  [^\t\n ]+ { wc++; cc+=yyleng;}
10 %%
11
12 int main(int argc ,char* argv[ ])
13 {
14     printf("Enter the input:\n");
15     yylex();
16     printf("The number of lines=%d\n",lc);
17     printf("The number of spaces=%d\n",sc);
18     printf("The number of words=%d\n",wc);
19     printf("The number of characters are=%d\n",cc);
20 }
21
22 int yywrap( )
23 {
24     return 1;
25 }
26
27
28
29
30
31
32 /*
33 ALGORITHM
34 1. Start
35 2. Initialize counter for lines 'lc', spaces 'sc', words 'wc' and characters 'cc' to zero
36 3. Define Lexical Rules:
37     3.1 [\n]: Matches newline characters, increments line count by 1
38     3.2 [ \t]: Matches spaces or tabs, increments space count by 1
39     3.3 [^\t\n ]: Matches sequences of characters that doesn't include spaces, tabs, newline,
40 indicating it is a word. Increments word count by 1
41 4. main() function:
42     4.1 Get user input
43     4.2 Invokes yylex() to perform lexical analysis
44     4.3 Prints the output
45 5. yywrap() function:
46     5.1 Called when input exhausted
47     5.2 Return 1 to show EOF
48 6. Stop
49 */
```