

Experiment -21

Develop a lexical Analyzer to ignore spaces , newlines , comments and words using lexical program.

Program:

```
%{  
#include <stdio.h>  
  
int nchar = 0, nword = 0, nline = 0;  
%}  
  
%%  
  
\n      { nline++; }  
[ \t]      { /* ignore spaces/tabs */ }  
[a-zA-Z0-9]+ { nword++; nchar += yyleng; }  
.       { nchar++; }  
%%  
  
int yywrap() { return 1; }  
  
int main() {  
    yylex();  
    printf("Number of characters = %d\n", nchar);  
    printf("Number of words    = %d\n", nword);  
    printf("Number of lines     = %d\n", nline);  
    return 0;  
}
```

Output:

```
C:\Windows\System32\cmd.e > + ^

C:\Compiler>dir
Volume in drive C is OS
Volume Serial Number is 2004-649C

Directory of C:\Compiler

28-08-2025  22:28      <DIR>          .
28-08-2025  21:47          115 17.c
28-08-2025  22:00          132 18.c
28-08-2025  22:13          126 19.html
28-08-2025  22:21          83 20.c
28-08-2025  22:28          72,245 a.exe
28-08-2025  21:26          420 EXP_16.l
28-08-2025  21:46          349 EXP_17.l
28-08-2025  22:00          389 EXP_18.l
28-08-2025  22:12          244 EXP_19.l
28-08-2025  22:20          267 EXP_20.l
28-08-2025  22:28          37,310 lex.yy.c
28-08-2025  22:28      <DIR>          OUTPUTS
28-08-2025  21:33          79 sample.c
12 File(s)          111,759 bytes
2 Dir(s)  112,000,958,464 bytes free

C:\Compiler>set path=C:\Program Files (x86)\GnuWin32\bin

C:\Compiler>flex EXP_21.l

C:\Compiler>set path=C:\Program Files\CodeBlocks\MinGW\bin

C:\Compiler>gcc lex.yy.c

C:\Compiler>a.exe < 21.txt
Number of characters = 50
Number of words      = 14
Number of lines       = 3

C:\Compiler>
```