Matthew Groover
Lab 1
1-22-20

I modified the original code by defining a new array that counts the amount of numbers that occur and is incremented when one or more numbers is found on a line. Using the regular expression [0-9]+ after that I added new parts to the formatted print so that it prints in the same way the word count is defined.

```
  /*  This lex routine uses a counting array to match alphabeticstrings
       and make a frequency count.

       The real item to notice is that yywrap() is called at EOF and then is run
       to do what we need to do.   yywrap() returns true when we have a successful
       end to the program.  We may want to return false (0) if we want to lexing process
       to fail

      Shaun Cooper
      January 2015

      added new ingo yadda yadda
  */
        int lgths[100];
      int lets[100];

%%
[a-zA-Z]+      lgths[yyleng]++;
[0-9]+         lets[yyleng]++;
.         |
\n        ;
%%

yywrap()
{
        int i;
        printf("Length  No.words   No.Numbers \n");
        for (i=1; i<100; i++) {
                if (lgths[i] > 0) {
                        printf("%5d%10d%10d\n",i,lgths[i],lets[i]);
                }
        }
        return(1);


}

main()
{  yylex();
```

//matthew groover
//1-22-20
//This makefile runs two commands on the same file
//first it updates lex.yy.c by compiling the new
//wordlengthlab1 then compiles both into a single executable
wordlength:wordlengthlab1.l lex.yy.c
	lex wordlengthlab1.l
	gcc -o wordlength lex.yy.c

```
mgroover@kahn:~/compilers> make
lex wordlengthlab1.l
gcc -o wordlength lex.yy.c
wordlengthlab1.l:25:1: warning: return type defaults to ‘int’ [-Wimplicit-int]

 ^~
wordlengthlab1.l:39:1: warning: return type defaults to ‘int’ [-Wimplicit-int]
 {  yylex();
 ^~~~
mgroover@kahn:~/compilers> ./wordlength < /etc/passwd
Length  No.words   No.Numbers
    1        42         6
    2         4         6
    3       108        61
    4        58         1
    5        40         6
    6        39         0
    7        53         0
    8         9         0
   10         7         0
   11         4         0
   12         1         0
   14         2         0
   15         1         0
mgroover@kahn:~/compilers>
```