# Count no of Vowels & Consonants

## Code:

vc.l - Notepad

File Edit Format View Help

```
%{
 #include<stdio.h>
 int v=0,c=0;
%}

%%
[aeiouAEIOU]          {v++;}
[a-zA-Z]              {c++;}
 .                    { }
%%
int yywrap(void) {
return 1;
}
int main()
{
  printf("Enter the Text:" );
  yylex();
  printf("\nNo. of Vowels=%d\nNo. of Consonents=%d\n",v,c);
  return 0;
}
```

## Output:

```
C:\Windows\System32\cmd.exe
C:\Users\Student\Desktop\SPCC D12B>vc.exe
Enter the Text:SPCC lab



No. of Vowels=1
No. of Consonents=6

C:\Users\Student\Desktop\SPCC D12B>
```

# Count no of Words, characters & lines

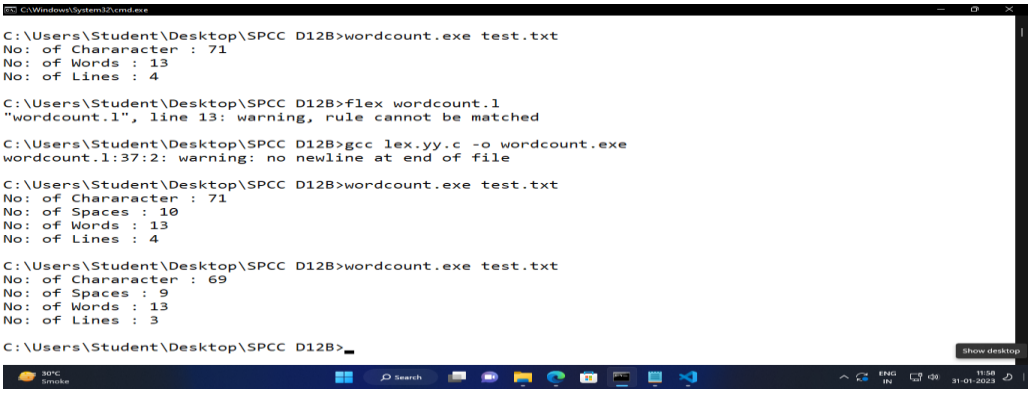
## Code:

wordcount.L - Notepad

File Edit Format View Help

```
%{
#include <stdlib.h>
#include <stdio.h>
int charCount=0;
int wordCount=0;
int spaceCount=0;
int lineCount=0;
%}
%%
\n {charCount++; lineCount++;}
[  \t] { spaceCount++; charCount+=yyleng;}
[^ \t\n]+ {wordCount++; charCount+=yyleng;}
. {charCount++;}
%%
int yywrap(void) {
return 1;
}
int main(int argc, char** argv)
{
if (argc > 1)
{
FILE *file;
file = fopen(argv[1], "r");
if (!file)
{
fprintf(stderr, "Could not open %s\n", argv[1]);
exit(1);
}
yyin = file;
}
yylex();
printf("No: of Chararacter : %d \n", charCount);
printf("No: of Spaces : %d \n", spaceCount);
printf("No: of Words : %d \n", wordCount);
printf("No: of Lines : %d \n", lineCount);
return 0;
}
```

## Output:



# Count no of keywords,identifiers & operators

## Code:

keyId.L - Notepad

File Edit Format View Help

```
%{
#include<stdio.h>
#include<stdlib.h>
int keywords =0;
int identifiers =0;
int operators =0;
%}

%%

if|else|while|do|for|int|printf|return|main    {keywords++;}
[a-zA-Z][a-zA-Z0-9]*     {identifiers++;}
\+|\-|\/|\%|\=|\!=     {operators++;}
.                ;
%%
int yywrap(void){
return 1;
}
int main(int argc, char** argv){
   if(argc > 1){
FILE *file;
file = fopen(argv[1],"r");
if(!file){
fprintf(stderr,"could not open %s\n",argv[1]);
```

keyId.L - Notepad

File Edit Format View Help

```
}
yyin = file;
}
yylex();
printf("keywords: %d\nIdentifiers: %d\nOperators:%d\n",keywords,identifiers,operators);
return 0;
}
```

porgram.c

```
#include<stdio.h>
int main(){
float num1;
double num2;

printf("enter a no :");
scanf("%f,&num1);
printf("enter another number");
scanf("%lf,@num2);
printf("num1= %f\n ",num1);
printf("num2= %f\n ",num2);
return 0;
}
```

## Output:

```
C:\Users\Student\Desktop\exp2>keywords.exe porgram.c new.txt




keywords: 7
Identifiers: 27
Operators:6
```

# Identify Even & odd integers

## Code:

eve_odd.L - Notepad

File Edit Format View Help

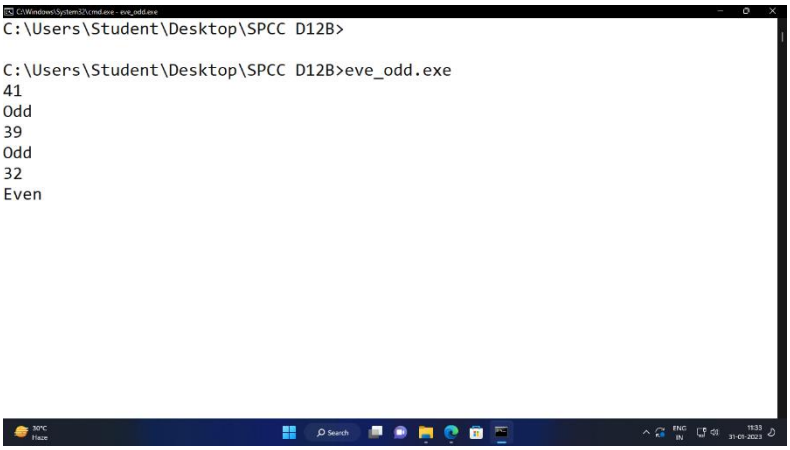
```
/*Lex program to take check whether
the given number is even or odd */

%{
#include<stdio.h>
int i;
%}

%%
[0-9]+      {i=atoi(yytext);
                    if(i%2==0)
                            printf("Even");
                    else
                    printf("Odd");}
%%
int yywrap(){}
/* Driver code */
int main()
{

            yylex();
            return 0;
}
```

## Output:



# Count of printf & scanf statements in C program

## Code:

printf.L - Notepad

File Edit Format View Help

```
%{

#include<stdio.h>

int pfc = 0, sfc = 0;

%}

%%
"printf" {fprintf(yyout,"writef"); pfc++;}
"scanf" {fprintf(yyout, "readf"); sfc++;}

%%
main(int argc, char *argv[])
{

if(argc != 3 )
{
printf("Usage: ./a.out in.txt out.txt\n");

`
printf("Usage: ./a.out in.txt out.txt\n");

yyin = fopen(argv[1], "r");
yyout = fopen(argv[2], "w");
yylex();
printf("\n the number of printf lines = %d\n", pfc);
printf("\n the number of scanf lines = %d\n", sfc);

exit(0);
}
}

int yywrap()

{
return 1;
}
```

inp_print - Notepad

File Edit Format View Help

```
#include<stdio.h>
int main(){
printf("Hello World");
int a;
printf("Enter number");
scanf("%d", &a);
printf("Your number is %d", a);
return 0;
}
```

## Output:

```
E:\spcc\exp 3\printf scanf>printf_scanf.exe in.txt
Usage: ./a.out in.txt out.txt
#include<stdio.h>

int main(){

writef("Hello World");
int a;
writef("Enter number");
readf("%d", &a);
writef("Your number is %d", a);
return 0;

}
 the number of printf lines = 3

 the number of scanf lines = 1
```

Classify English words as verbs, adverbs, adjectives etc.

## Code:

EngWords - Notepad

File Edit Format View Help

```
%{
#include <stdio.h>
#include <string.h>
int v_count = 0;
int n_count = 0;
int adv_count = 0;
int adj_count = 0;
%}

%x v_token
%x n_token
%x adj_token
%x adv_token

%%

"run" {
v_count++;
BEGIN(v_token);
}

"dog"|"book"|"computer"|"music"|"food"|"television"|"tree"|"house"|"car"|"phone"|"desk"|"chair"|"water"|"person"|"table"|"r
n_count++;
BEGIN(n_token);
```

EngWords - Notepad

File Edit Format View Help

```
adj_count++;
BEGIN(adj_token);
}

"quickly"|"slowly"|"loudly"|"softly"|"happily"|"sadly"|"smoothly"|"roughly" {
adv_count++;
BEGIN(adv_token);
}

<v_token>.|\n { BEGIN(INITIAL);}
<n_token>.|\n { BEGIN(INITIAL);}
<adv_token>.|\n { BEGIN(INITIAL);}
<adj_token>.|\n { BEGIN(INITIAL);}

. { /* ignore all other characters */}

%%
int yywrap(){}
int main(int argc, char *argv[])
{
yylex();
printf("number of verbs: %d\n", v_count);
printf("number of nouns: %d\n", n_count);
printf("number of adjectives: %d\n", adj_count);
```

## Output:

```
C:\Users\Student\Desktop\spcc>flex naav.l

C:\Users\Student\Desktop\spcc>gcc lex.yy.c -o naav.exe
naav.l:54:2: warning: no newline at end of file

C:\Users\Student\Desktop\spcc>naav.exe input.txt
the dog is very happy as he run quickly but sometimes it is sad when it sees the table on the road and it is also hungry
 for apple.

^Z
number of verbs: 1
number of nouns: 3
number of adjectives: 2
number of adverbs: 1
```