Compiler Design

Assignment – 3: Implement below code:

1) Generate Histogram of Words:

CODE:

```
%option noyywrap
%{
        #include<stdio.h>
        #include<string.h>
%}
%%
%%
char *text;
int main()
{

text=(char *)malloc(100*sizeof(char));
char *word=(char *)malloc(10*sizeof(char));
char *temp=(char *)malloc(10*sizeof(char));
printf("\nEnter the text of max length 99 and max length of any word
should be 10\n");
gets(text);
int len=0;
int words=0;
int unique=0;
while(len<strlen(text))
{
words++;
sscanf(text+len,"%s",word);
int ctr=0;
int freq=0;
```

```
while(ctr<strlen(text))
{
sscanf(text+ctr,"%s",temp);
if(strcmp(word,temp)==0)
freq++;
ctr=ctr+strlen(temp)+1;
}
if(freq!=1)
printf("\n<%s> is not unique and has frequency %d\n",word,freq);
else
{
   printf("\nThe word <%s> is unique\n",word);
   unique++;
}
len=len+strlen(word)+1;
}
printf("\nThe number of unique words is %d\n",unique);
printf("\nThe total number of words are %d\n",words);
getch();
}
```

Output:

```
C:\Flex Windows\EditPlusPortable\ASSIGNMENT\lab2\hisstt>cc

Enter the text of max length 99 and max length of any word should be 10
My name is arjun and My surname is vankani.

<My> is not unique and has frequency 2

The word <name> is unique

<is> is not unique and has frequency 2

The word <arjun> is unique

The word <and> is unique

<My> is not unique and has frequency 2

The word <surname> is unique

<is> is not unique and has frequency 2

The word <vankani.> is unique

The number of unique words is 5

The total number of words are 9
```

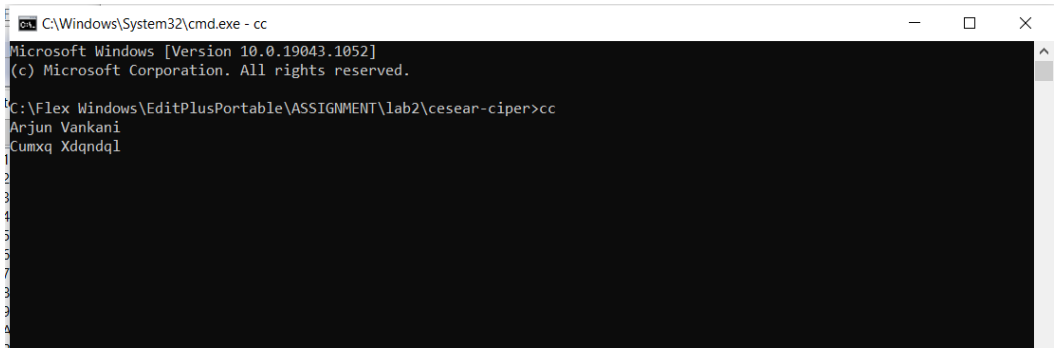
2) Cesar Cipher

CODE:

```
%option noyywrap
%{
#include<stdio.h>
%}

%%
[a-z] {char ch = yytext[0];
ch += 3;
if (ch> 'z') ch -= ('z'+1- 'a');
printf ("%c" ,ch );
}
[A-Z] { char ch = yytext[0] ;
ch += 2;
if (ch> 'Z') ch -= ('Z'+1- 'A');
printf("%c",ch);
}
%%
int main()
{
 yylex();
 return 0;
}
```

OUTPUT:



3) Count Comment:

CODE:

```
%option noyywrap
%{
   #include<stdio.h>
   #include<stdlib.h>
   int a=0,b=0,c=0,d;
%}
%%
"//".* {a++;}
"/*" {b++;}
.*"*/" {b--;c++;}
%%
void main(int argc,char *argv[]){
   yyin=fopen(argv[1],"r");
   yylex();
   printf("\nSingle line %d \nMultiline %d \n",a,c);
   d=a+c;
   printf("-----------------------");
   printf("\nTotal %d \n",d);
}
```

OUTPUT: