

Name	Registration Number
1. Muema Stephen Nyamai	SCT211-0081/2016
2. Aphonse Kiprop Kipng'etich	SCT211-0083/2016
3. Mburu Francis Kimani	SCT211-0058/2016
4. Mirugah Brodrick	SCT211-0093/2016
5. Oyugi Javan	SCT211-2078/2015
6. Okoth Michelle	SCT211-0096/2016
7. Nyambura Diana	SCT211-0316/2016
8. Rahab Nderitu	SCT211-0302/2016
9. Catherine Karanja	SCT211-5240/2015

1. Write a LEX specification files to:

i). Count the number of words in a file and their total size

```
%{
#include <stdio.h>
int words = 0, characters = 0, digits = 0, symbols = 0, size = 0;
}%

%%

[\\t \\n] words++;
[a-zA-Z] characters++;
[0-9] digits++;
. symbols++;
%%

int main(){
    yyin = fopen("text.txt", "r");
    yylex();
    size = characters + digits + symbols;

    printf("The number of words is: %d \\n", words);
    printf("There are %d characters, %d digits and %d symbols \\n",
characters, digits, symbols);
    printf("The total size is: %d \\n", size);
    fclose(yyin);
    return 0;
}

int yywrap(){
    return 1;}
}
```

ii). Accepts the English language words (without bothering for the meaning) and replaces each occurrence of the string "abc" in it to "ABC".

```
%{
#include <stdio.h>
#include <string.h>

char match[] = "abc", replace_match[] = "ABC";

}%

%%
abc {fprintf(yyout, "%s", replace_match);}
. {fprintf(yyout, "%s", yytext);}
%%

int main(){
yyin = fopen("input.txt", "r");
yyout = fopen("output.txt", "w");
yylex();
fclose(yyin);
fclose(yyout);
return 0;
}
int yywrap(){
return 1;}
}
```

2. The following is a listing of a set of verbs:

```
is am are were
was be being been
do does did will
would should can could
has have had go
```

Write a simple LEX specification to recognize these verbs

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

%%
[\\t]+
is |
am |
are |
was |
be |
being |
do |
does |
did |
will |
would |
should |
can |
could |
has |
have |
had |
go {printf("%s: is a verb\\n", yytext);}
[a-zA-Z]+ {printf("%s: is not a verb\\n", yytext);}
%%
int main(){
    yylex();
    return 0;
}
int yywrap(){
    return 1;
}
```