

Experiment 4

Lex

Q1

```
%{
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <ctype.h>
%}

%%

"int" | "float" | "char" | "double" | "void" { printf("Keyword: %s\n", yytext); }
[a-zA-Z_][a-zA-Z0-9_]* { printf("Identifier: %s\n", yytext); }
[0-9]+[.][0-9]+ { printf("Float Number: %s\n", yytext); }
[0-9]+ { printf("Integer Number: %s\n", yytext); }
. { printf("Other: %s\n", yytext); }

%%

int yywrap(void) {}

int main(int argc, char *argv[]) {
    if (argc < 2) {
        printf("Usage: %s <filename>\n", argv[0]);
        return 1;
    }

    FILE *file = fopen(argv[1], "r");
    if (file == NULL) {
```

```
    printf("Unable to open the file.\n");  
    return 1;  
}  
  
yyin = file;  
yylex();  
  
fclose(file);  
  
return 0;  
}
```

```

└─$ cat cfile.c
int a = 12;
int b = "as as";
float x= 121.12;
int ahsglas = 1221;

└─(kali1@kali)-[~/@1_DDrive/Code_Files]
└─$ lex first.l

└─(kali1@kali)-[~/@1_DDrive/Code_Files]
└─$ gcc lex.yy.c

└─(kali1@kali)-[~/@1_DDrive/Code_Files]
└─$ ./a.out cfile.c
Keyword: int
Other:
Identifier: a
Other:
Other: =
Other:
Integer Number: 12
Other: ;

Keyword: int
Other:
Identifier: b
Other:
Other: =
Other:
Other: "
Identifier: as
Other:
Identifier: as
Other: "
Other: ;

Keyword: float
Other:
Identifier: x
Other: =
Other:
Float Number: 121.12
Other: ;

Keyword: int
Other:
Identifier: ahsglas
Other:
Other: =
Other:
Integer Number: 1221
Other: ;

└─(kali1@kali)-[~/@1_DDrive/Code_Files]
└─$
```

Q2

```
%{
```

```
#include <stdio.h>
```

```
#include <string.h>
```

```
#include <ctype.h>
```

```
int i = 0;
```

```
int vowelCount = 0;
```

```
int wordCount = 0;
```

```
%}
```

```
%%
```

```
[a-zA-Z]+ {
```

```
    vowelCount = 0;
```

```
    for (int j = 0; j < yytext[j]; j++) {
```

```
        char c = tolower(yytext[j]);
```

```
        if (c == 'a' || c == 'e' || c == 'i' || c == 'o' || c == 'u') {
```

```
            vowelCount++;
```

```
        }
```

```
    }
```

```
    if (vowelCount == 2) {
```

```
        printf("Word with exactly 2 vowels: %s\n", yytext);
```

```
        wordCount++;
```

```
    }
```

```
}
```

```
"\n" {
```

```
    printf("Number of words: %d\n", i);
```

```
    printf("Number of words with exactly 2 vowels: %d\n", wordCount);
```

```
    i = 0;
```

```
    wordCount = 0;
```

```
}
```

```
. { }
```

```
%%
```

```
int yywrap(void) {}
```

```
int main()
{
    printf("Enter the filename: ");
    char filename[100];
    scanf("%s", filename);

    FILE* file = fopen(filename, "r");
    if (file == NULL) {
        printf("Unable to open the file.\n");
        return 1;
    }

    yyset_in(file);

    printf("Analyzing the file...\n");
    yylex();

    fclose(file);
    return 0;
}
```

```

(kali1@kali)-[~/@1_DDrive/Code_Files]
└─$ cat cfile.c
int a = 12;
int b = "as as";
float x= 121.12;
int ahsg1as = 1221;

(kali1@kali)-[~/@1_DDrive/Code_Files]
└─$ lex first.l

(kali1@kali)-[~/@1_DDrive/Code_Files]
└─$ gcc lex.yy.c

(kali1@kali)-[~/@1_DDrive/Code_Files]
└─$ ./a.out cfile.c
Enter the filename: cfile.c
Analyzing the file...
Number of words: 0
Number of words with exactly 2 vowels: 0
Number of words: 0
Number of words with exactly 2 vowels: 0
Word with exactly 2 vowels: float
Number of words: 0
Number of words with exactly 2 vowels: 1
Number of words: 0
Number of words with exactly 2 vowels: 0
Number of words: 0
Number of words with exactly 2 vowels: 0

(kali1@kali)-[~/@1_DDrive/Code_Files]
└─$

```

Q3

```

%{

#include <stdio.h>

#include <string.h>

#include <ctype.h>

int i = 0;

int startWithTCount = 0;

int endWithTCount = 0;

%}

/* Rules Section */

%%

```

```
[t][a-zA-Z]* {  
    printf("Word starting with 't': %s\n", yytext);  
    startWithTCount++;  
}
```

```
[a-zA-Z]*[T] {  
    printf("Word ending with 'T': %s\n", yytext);  
    endWithTCount++;  
}
```

```
"\n" {  
    printf("Number of words starting with 't': %d\n", startWithTCount);  
    printf("Number of words ending with 'T': %d\n", endWithTCount);  
    startWithTCount = 0;  
    endWithTCount = 0;  
}
```

```
. { }
```

```
%%
```

```
int yywrap(void) {}
```

```
int main()  
{  
    printf("Enter the input: ");  
    yylex();  
    return 0;  
}
```

```

(kali㉿kali)-[~/@1_DDrive/Code_Files]
└─$ lex first.l

(kali㉿kali)-[~/@1_DDrive/Code_Files]
└─$ gcc lex.yy.c

(kali㉿kali)-[~/@1_DDrive/Code_Files]
└─$ ./a.out
Enter the input: hello hi bye
Number of words: 0
Number of words starting with 't': 0
Number of words ending with 'T': 0
^[[A^[[B^C

(kali㉿kali)-[~/@1_DDrive/Code_Files]
└─$ lex first.l

(kali㉿kali)-[~/@1_DDrive/Code_Files]
└─$ gcc lex.yy.c

(kali㉿kali)-[~/@1_DDrive/Code_Files]
└─$ ./a.out
Enter the input: hello that is my BaT yu rae out that is wrong whaT
Word starting with 't': that
Word ending with 'T': BaT
Word starting with 't': t
Word starting with 't': that
Word ending with 'T': whaT
Number of words starting with 't': 3
Number of words ending with 'T': 2
$

```

Q4

```
%{
```

```
#include <stdio.h>
```

```
#include <string.h>
```

```
#include <ctype.h>
```

```
int i = 0;
```

```
int mixedCaseCount = 0;
```

```
%}
```

```
/* Rules Section */
```

```
%%
```

```
[a-zA-Z]+ {
```

```
    int hasLower = 0;
```

```
    int hasUpper = 0;
```



```

for (int j = 0; j < yytext[j]) {
    if (islower(yytext[j])) {
        hasLower = 1;
    } else if (isupper(yytext[j])) {
        hasUpper = 1;
    }
}

if (hasLower && hasUpper) {
    printf("Word in mixed case: %s\n", yytext);
    mixedCaseCount++;
}
}

"\n" {
    printf("Number of words: %d\n", i);
    printf("Number of words in mixed case: %d\n", mixedCaseCount);
    i = 0;
    mixedCaseCount = 0;
}

. { }

%%

int yywrap(void) {}

int main()
{
    printf("Enter the input: ");
    yylex();
    return 0;
}

```

}

```
(kali1@kali)-[~/@1_DDrive/Code_Files]
$ lex first.l

(kali1@kali)-[~/@1_DDrive/Code_Files]
$ gcc lex.yy.c

(kali1@kali)-[~/@1_DDrive/Code_Files]
$ ./a.out
Enter the input: hellow how areY ou doiNG to study
Word in mixed case: hellow
Word in mixed case: areY
Word in mixed case: doiNG
Number of words in mixed case: 3
```