



SYMBIOSIS INSTITUTE OF TECHNOLOGY, PUNE

Symbiosis International (Deemed University)

(Established under section 3 of the UGC Act, 1956)

Re-accredited by NAAC with 'A' grade (3.58/4) | Awarded Category – I by UGC

Founder: Prof. Dr. S. B. Mujumdar, M. Sc., Ph. D. (Awarded Padma Bhushan and Padma Shri by President of India)

Assignment No. 06

Subject:	Compiler Construction Lab
Name of Student	Onkar Mendhapurkar
PRN No.	22070122135
Branch	CSE B2, Batch (2022-26)
Academic Year & Semester	2022-26
Date of Performance	04/09/2025
Title of Assignment:	Lex Program
Practice Questions	<ol style="list-style-type: none">Lex program to count lines ending with "com".Lex program to count URLs ending with "org". <p>Practice question below</p> <ol style="list-style-type: none">Lex program to count URLs ending with "edu".
Source Code	<pre>1. %{ #include <stdio.h> int com_count = 0; %} %% .*com\n { com_count++; } // match entire line ending with com</pre>

```
.*\n      ; // ignore other lines\n\n%%\n\nint yywrap() { return 1; }\n\nint main() {\n    yylex();\n    printf("Number of lines ending with 'com': %d\n", com_count);\n    return 0;\n}
```

2.

```
%{\n#include <stdio.h>\nint org_count = 0;\n%}\n\n%%
```

```
(https?:\/\/)[a-zA-Z0-9._-]*\.org([ \t\n]|\/) { org_count++; }
```

```
.\\n ; // ignore everything else
```

```
%%\n\nint yywrap() { return 1; }
```

```
int main() {\n    yylex();\n    printf("Number of URLs ending with 'org': %d\n", org_count);\n    return 0;\n}
```

3.

```
%{\n#include <stdio.h>\nint edu_count = 0;\n%}\n\n%%
```

```
(https?:\/\/)[a-zA-Z0-9._-]*\.edu([ \t\n]|\/) { edu_count++; }
```

```

.|n ; // ignore everything else

%%

int yywrap() { return 1; }

int main() {
    yylex();
    printf("Number of URLs ending with 'edu': %d\n", edu_count);
    return 0;
}

```

Output Screenshot

1.

```

battlemachine@DESKTOP-FU1975B:/mnt/c/Users/DELL/CCL$ cd Assign6
battlemachine@DESKTOP-FU1975B:/mnt/c/Users/DELL/CCL$ nano count_com.l
battlemachine@DESKTOP-FU1975B:/mnt/c/Users/DELL/CCL/Assign6$ cat count_com.l
%{
#include <stdio.h>
int com_count = 0;
%}

%*
.*com\n    { com_count++; } // match entire line ending with com
.*\n        ; // ignore other lines
%*

int yywrap() { return 1; }

int main() {
    yylex();
    printf("Number of lines ending with 'com': %d\n", com_count);
    return 0;
}
battlemachine@DESKTOP-FU1975B:/mnt/c/Users/DELL/Assign6$ nano input.txt
battlemachine@DESKTOP-FU1975B:/mnt/c/Users/DELL/CCL/Assign6$ lex count_com.l
battlemachine@DESKTOP-FU1975B:/mnt/c/Users/DELL/CCL/Assign6$ cc lex.yy.c -o count_com
battlemachine@DESKTOP-FU1975B:/mnt/c/Users/DELL/CCL/Assign6$ ./count_com < input.txt
Number of lines ending with 'com': 2
battlemachine@DESKTOP-FU1975B:/mnt/c/Users/DELL/CCL/Assign6$ |

```

2.

```

battlemachine@DESKTOP-FU1975B:/mnt/c/Users/DELL/CCL/Assign6$ nano count_org.l
battlemachine@DESKTOP-FU1975B:/mnt/c/Users/DELL/CCL/Assign6$ cat count_org.l
%{
#include <stdio.h>
int org_count = 0;
%}

%*
(https?:\/\/)[a-zA-Z0-9._-]*\.org([ \t\n]|\/)    { org_count++; }
.|\\n        ; // ignore everything else
%*

int yywrap() { return 1; }

int main() {
    yylex();
    printf("Number of URLs ending with 'org': %d\n", org_count);
    return 0;
}
battlemachine@DESKTOP-FU1975B:/mnt/c/Users/DELL/CCL/Assign6$ cat input.txt
www.google.com
https://wikipedia.org
http://example.org
https://mit.edu
randomtext.com
http://harvard.edu
battlemachine@DESKTOP-FU1975B:/mnt/c/Users/DELL/CCL/Assign6$ lex count_org.l
battlemachine@DESKTOP-FU1975B:/mnt/c/Users/DELL/CCL/Assign6$ cc lex.yy.c -o count_org
battlemachine@DESKTOP-FU1975B:/mnt/c/Users/DELL/CCL/Assign6$ ./count_org < input.txt
Number of URLs ending with 'org': 2

```

3.

```

battlemachine@DESKTOP-FU1975B:/mnt/c/Users/DELL/CCL/Assign6$ nano count_edu.l
battlemachine@DESKTOP-FU1975B:/mnt/c/Users/DELL/CCL/Assign6$ cat count_edu.l
%{
#include <stdio.h>
int edu_count = 0;
%}

<% (https?:\/\/)[a-zA-Z0-9./_-]*\.edu([ \t\n]|\/) { edu_count++; }

.|\n ; // ignore everything else
%%

int yywrap() { return 1; }

int main() {
    yylex();
    printf("Number of URLs ending with 'edu': %d\n", edu_count);
    return 0;
}
battlemachine@DESKTOP-FU1975B:/mnt/c/Users/DELL/CCL/Assign6$ cat input.txt
www.google.com
https://wikipedia.org
http://example.org
https://mit.edu
randomtext.com
http://harvard.edu
battlemachine@DESKTOP-FU1975B:/mnt/c/Users/DELL/CCL/Assign6$ lex count_edu.l
battlemachine@DESKTOP-FU1975B:/mnt/c/Users/DELL/CCL/Assign6$ cc lex.yy.c -o count_edu
battlemachine@DESKTOP-FU1975B:/mnt/c/Users/DELL/CCL/Assign6$ ./count_edu < input.txt
Number of URLs ending with 'edu': 2
battlemachine@DESKTOP-FU1975B:/mnt/c/Users/DELL/CCL/Assign6$ |

```

Conclusion	These LEX programs show how simple rules can automate number system conversions. They highlight the efficiency of lexical analysis in scanning, recognizing patterns, and transforming input seamlessly.
------------	--