

# CSE 358 Assignment 4

Arnav Jain - 2200002018

## Question 1

LEX is a lexical analyzer generator used in compiler construction. It converts a set of regular expressions into a finite state machine to recognize patterns in input text, producing tokens for further processing by a process.

### Structure of LEX program:

A LEX program consists of 3 sections:

- 1) Definition section: Defines macros and header files.
- 2) Rules section: Contains pattern-action pairs where regular expressions define patterns and actions specify corresponding code.
- 3) User code section: Includes additional C code usually for functions needed in actions.

The LEX tool generates a C program (lex.yy.c), which when compiled produces a lexical analyser (a.out) to process input text.

## Question 2

Code:

```
≡ q2.l  ×
≡ q2.l
1  /*lex program to count number of words*/
2  %{
3  #include<stdio.h>
4  #include<string.h>
5  int i = 0;
6  %}
7
8  /* Rules Section*/
9  %%
10 ([a-zA-Z0-9])* {i++;} /* Rule for counting number of words*/
11
12 "\n" {printf("%d\n", i); i = 0;}
13 %%
14
15 int yywrap(void){}
16
17 int main()
18 {
19     // The function that starts the analysis
20     yylex();
21
22     return 0;
23 }
24
25
26 // commands:
27 // lex -o q2.c q2.l
28 // gcc q2.c -ll -o q2
29 // ./q2
30
```

Result:



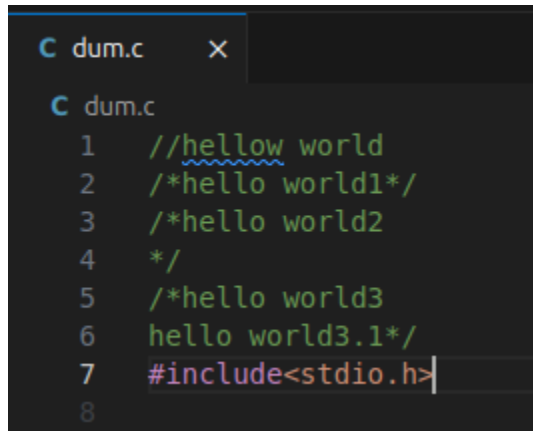
The screenshot shows a terminal window within a code editor. The terminal title bar includes tabs for PROBLEMS (2), OUTPUT, DEBUG CONSOLE, TERMINAL (active), PORTS, POLYGLOT NOTEBOOK, GITLENS, SPELL CHECKER (2), and COMMENTS. The terminal content shows a prompt for a user named arnav at a specific location, followed by the execution of a program named q2. The program outputs three lines: 'hi im arnav', '3', and 'hello.im.arnav', followed by a prompt for input. The input '.3' is entered, and the cursor is positioned at the end of the line.

```
arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiler-Techniques/LAB 4$ ./q2
hi im arnav
3
hello.im.arnav
..3

```

### Question 3

File used for testing:

A screenshot of a code editor window titled 'C dum.c' with a close button 'X'. The editor shows the following C code:

```
C dum.c
1 //hellow world
2 /*hello world1*/
3 /*hello world2
4 */
5 /*hello world3
6 hello world3.1*/
7 #include<stdio.h>
8
```

Code:

```
q3.l x
q3.l
2  #include<stdio.h>
3  #include<stdlib.h>
4  int a=0,b=0,c=0,d;
5  %}
6  %x COMMENT
7  /* an exclusive state that does not also match normal stuff */
8  %%
9  "//".* {a++;}
10 "/*" { BEGIN COMMENT; }
11 <COMMENT>"*/" {c++; BEGIN INITIAL; }
12 <COMMENT>. ;
13 %%
14 void main(int argc,char *argv[]){
15     yyin=fopen(argv[1],"r");
16     yylex();
17     printf("single line %d \nmultiline %d \n",a,c);
18     d=a+c;
19     printf("total %d \n",d);
20 }
21
22 // commands:
23 // lex -o q3.c q3.l
24 // gcc q3.c -ll -o q3
25 // ./q3 dum.c
```

Result:

```
PROBLEMS 4 OUTPUT DEBUG CONSOLE TERMINAL PORTS POLYGLOT NOTEBOOK GITLENS SPELL CHECKER 4 COMMENTS
arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiler-Techniques/LAB 4$ ./q3 dum.c

#include<stdio.h>
single line 1
multiline 3
total 4
arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiler-Techniques/LAB 4$
```

**For code refer GitHub**

<https://github.com/arnavjain2710/Compiler-Techniques/tree/main/LAB%204>