

INDEX

Sr. No.	Experiment Description	Experiment Date	Submission Date	Remarks / Signature
5.)	Design a program for creating a machine that accepts the string always ending with 0's.	30/02/24	06/02/24	W.P. 1st
6.)	Design a program for Moore Machine	30/02/24	06/02/24	W.P. 1st
7.)	Design a program to find 1's complement using Mealy Machine	06/02/24	13/02/24	{ W.P. 1st }
8.)	Design a program to find 1's complement using Moore Machine	06/02/24	13/02/24	{ W.P. 1st }
9.)	Write a LEX program to count the no. of vowels, consonants and digits.	13/02/24	20/02/24	{ W.P. 1st }
10.)	To get echo of a string	13/02/24	20/02/24	
11.)	To count and recognise the keywords and identifiers	20/02/24	27/02/24	
12.)	To find words begin and ending with &	20/02/24	27/02/24	

Page : _____

Date : _____

INDEX

Sr. No.	Experiment Description	Experiment Date	Submission Date	Signature
13.)	Write a LEX program to identify the capital letters, small letters, digits & symbols.	27/02/24	5/03/24	1
14.)	Write a LEX program to check valid email id	27/02/24	5/03/24	
15.)	Write a LEX program to check valid mobile number	27/02/24	5/03/24	
16.)	Design a PDA for grammar "a^n b^n" for n>5	27/02/24	9/04/24	
17.)	Design a turing machine for m+n where m and n are unary	31/04/24	16/05/24	

LAB-1

- AIM: WAP to count number of characters, whitespaces and line by reading a text file in C.

→ #include <stdio.h>
 # include <stdlib.h>
 # include <string.h>

int main ()
{

FILE * fileptr;
 char ch;
 int word = 0;
 int whitespace = 0;
 int line = 0;
 fileptr = fopen ("text.txt", "r");
 if (fileptr == NULL) {
 printf ("no file");
 }

do {

ch = fgetc (fileptr);
 if (ch == ' ') {
 whitespace++;
 }
 if (fgetc (fileptr) == '\n') word--;

else if (ch == '\n') {
 line++;
 word++;
 }

}

while ($ch \neq EOF$):

word++;

line++;

word = word + whitespace;

char (fileptr);

printf (" %d %d %d ", word, whitespace, line);

return 0;

3

ul 1234

test.out :-

day

this is the first line

L>6 4 2

text-file.txt :-

hey

this is the first lab

↳ vowel = 6 consonant = 14 word = 6 whitespace = 4
line = 2

- ① AIM - WAP to count number of vowels and consonants by reading a text file in C.

→ #include <stdio.h>
#include <stdlib.h>
#include <string.h>

int main () {

```
File * fileptr;
char ch;
int word = 0;
int whitespace = 0;
int line = 0;
int vowel = 0;
int consonant = 0;
fileptr = fopen ("text.txt", "r");
if (fileptr == NULL) {
    printf ("nofile");
}
```

do {

```
ch = fgetc (fileptr);
if (ch == ' ') {
    whitespace++;
}
```

```
else if (ch == '\n') {
    line++;
    word++;
}
```

```
else if ((ch >= 'a' && ch <= 'z') || (ch >= 'A' && ch <= 'Z'))
```

```
{ if (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u' ||  
     ch == 'A' || ch == 'E' || ch == 'I' || ch == 'O' || ch == 'U') {  
    vowel++;  
}
```

}

```
else {
```

```
    consonant++;  
}
```

}

```
while (ch != FOR);
```

```
word++;
```

```
line++;
```

```
word = word + whitespace;
```

```
fclose(fileptr);
```

```
printf ("vowel = %d consonant = %d word = %d  
whitespace = %d line = %d", vowel, consonant,  
word, whitespace, line);
```

```
return 0;
```

W
W
W

LAB - 2

file.txt :-

Hello, I am Adarsh. I study in Manipal University.
Manipal is a very good college. The students of
Manipal like to miss classes.

Thanks,
Adarsh.

→ Total Keywords Found : 5

① AIM: WAP to check whether the given string is keyword or not by reading a file in C

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

```
int isKeyword (char word[]) {
    char Keywords [] = {"Adarsh", "Manipal"};
```

```
for (int i=0; i<2; ++i) {
    if (strcmp (Keywords [i], word) == 0) {
        return 1;
    }
}
```

return 0;

```
int main () {
    char filename [100];
    char word [20];
    int count = 0;
```

~~FILE *file = fopen ("text.txt", "r");~~

```
if (file == NULL) {
    printf ("Error opening file! \n");
    return 0;
}
```

```
while (fscanf(file, "%s", word) != EOF) {  
    if (isKeyword(word)) {  
        count++;  
    }  
}
```

fclose(file);

printf("Total Keywords found: %d\n", count);

return 0;

upm
221

Enter Production String:

$$\cdot A \rightarrow xyz$$

\Rightarrow Type - 2 Production

$$\cdot A \rightarrow Bxz$$

\Rightarrow Type - 3 Production

$$\cdot ABz \rightarrow *$$

\Rightarrow Type - 0 production

$$\cdot abcde \rightarrow \Sigma$$

\Rightarrow Not a valid Production

LAB - 3

CLASSMATE
Date : 23/1/24
Page : 7

- ① AIM: WAP to check whether the entered string belongs to a Chomsky Hierarchy or not. If not then specify the type of grammar.

\rightarrow import java.util.*;

```
public class Main {
```

```
    public static void main (String [] args) {
```

~~Scanner s = new Scanner (System.in);
s.nextLine ("Enter Production String :");
String str = s.nextLine ();~~~~String [] arr = str.split ("-", 2);
int t0=0, t1=0, t2=0, t3=0;~~

```
for (int i=0; i < arr[0].length(); i++) {
```

~~char c = arr[0].charAt(i);~~~~int ascii = (int) c;~~~~if (ascii >= 65 && ascii <= 91) {~~~~t0=1;~~~~break;~~

3

```
if (arr[0].length() <= arr [1].length ()) {  
    t1=1;  
}
```

```
if (arr[0].length == 1) {
    t2 = 1;
}
```

```
int flag = 0;
for (int i = 0; i < arr[1].length(); i++) {
    char c = arr[1].charAt(i);
    int ascii = (int) c;
    if (ascii >= 65 && ascii <= 91) {
        flag++;
    }
}
```

```
if (flag == 0) {
    t3 = 0;
}
```

```
else if (flag == 1) {
```

```
if (arr[1].charAt(0) < 65 && arr[1].charAt(0) > 91)
    && (arr[1].charAt(arr[1].length() - 1) < 65 &&
        arr[1].charAt(arr[1].length() - 1) > 91)) {
```

```
t3 = 0;
```

```
else {
```

```
t3 = 1;
```

```
}
```

```
if ( $t_0 == 1$ ) {  
    if ( $t_1 == 1$ ) {  
        if ( $t_2 == 1$ ) {  
            if ( $t_3 == 1$ ) {  
                Sopln("Type-3 Production");  
            }  
        }  
    }  
}
```

else {

```
    Sopln("Type-2 Production");  
}  
}
```

else {

```
    Sopln("Type-1 Production");  
}  
}
```

else {

```
    Sopln("Type-0 Production");  
}  
}
```

else {

```
    Sopln("Not a valid production");  
}  
}
```

Wf2011

LAB-4

- ① AIM: Design a program for creating machine that accepts three consecutive one.

→ import java.util.*;

```
public static void main (String [] args) {
    Scanner myObj = new Scanner (System.in);
    System.out.println ("Enter the Binary String :");
    String str = myObj.nextLine();
```

```
String state = "q1";
for (int i = 0; i < str.length(); i++) {
    char ch = str.charAt(i);
    String c = String.valueOf(ch);
```

```
switch (state) {
```

```
case "q1":
```

```
if (c.equals ("0")) {
```

```
state = "q1";
```

```
} else if (c.equals ("1")) {
```

```
state = "q0";
```

```
}
```

```
break;
```

```
case "q0":
```

```
if (c.equals ("0")) {
```

```
state = "q1";
```

```
} else if (c.equals ("1")) {
```

```
state = "q1";
```

```
} break;
```

Output :-

Enter the Binary String:

101011
String rejected

1110101
String Accepted

111
String Accepted

10101110

String Accepted

101101

String rejected

case "q₂":

```

if (c.equals ("0")) {
    state = "q1";
}
else if (c.equals ("1")) {
    state = "q2";
}
break;
}
```

Case "q₂":

```

if (c.equals ("0")) {
    state = "q2";
}
else if (c.equals ("1")) {
    state = "q2";
}
break;
}
```

System.out.println ("State: "+state);

if (state.equals ("q₂")) {

System.out.println ("Your String is accepted");

else {

System.out.println ("String Rejected");

Up
Down
Left
Right

Output:

Enter the binary String:

101
String Accepted

1011
String Rejected

101010110

String Rejected

0110101101

String Accepted

LAB-5

- AIM: Design a program for creating machine that accepts the string always ending with 101.
→ import java.util.*;

```
public class Main {
    public static void main (String [] args) {
        Scanner obj = new Scanner (System.in);
        System.out.println ("Enter the Binary String:");
        String str = obj.nextLine();
        String state = "q1";
        for (int i=0; i< str.length(); i++) {
            char ch = str.charAt (i);
            String c = String.valueOf (ch);
            switch (state) {
                case "q1":
                    if (c.equals ("0")) {
                        state = "q1";
                    } else if (c.equals ("1")) {
                        state = "q0";
                    }
                    break;
                case "q0":
                    if (c.equals ("0")) {
                        state = "q1";
                    } else if (c.equals ("1")) {
```

switch (state) {
case "q1":

```
if (c.equals ("0")) {
    state = "q1";
} else if (c.equals ("1")) {
    state = "q0";
}
break;
```

case "q0":

```
if (c.equals ("0")) {
    state = "q1";
} else if (c.equals ("1")) {
```

3 State = "q0";

break;

case "q1":

if (c.equals ("0")) {
State = "q1";

} else if (c.equals ("1")) {

State = "q2";

}

break;

case "q2":

if (c.equals ("0")) {
State = "q1";

} else if (c.equals ("1")) {

State = "q0";

}

break;

}

System.out.println ("State: " + state);

if (state.equals ("q2")) {

System.out.println ("Your String is accepted");

} else {

System.out.println ("String Rejected");

}

}

Output:

Enter the binary string:

110

String Accepted

0

String Accepted

11 11

String Accepted

1010101

String Rejected

LAB-6

- AIM: Design a program for Mad 3 machine
→ import java.util.*;

```
public class Main {  
    public static void main (String [] args) {  
        Scanner sc = new Scanner (System.in);  
        System.out.println ("Enter the Binary String");  
        String str = sc.nextLine ();
```

```
        String state = "q";  
        for (int i = 0; i < str.length(); i++) {  
            char ch = str.charAt (i);  
            String c = String.valueOf (ch);
```

```
            switch (state) {  
                case "q":
```

```
                    if (c.equals ("0")) {  
                        state = "q0";
```

```
                } else if (c.equals ("1")) {  
                    state = "q1";
```

```
                }  
                break;
```

```
            case "q0":
```

```
                if (c.equals ("0")) {  
                    state = "q0";
```

```
                } else if (c.equals ("1")) {  
                    state = "q1";
```

```
                }  
                break;
```

case "q₁":

if (c.equals ("0")) {

state = "q₂";

} else if (c.equals ("1")) {

state = "q₂";

}

break;

Case "q₂":

if (c.equals ("0")) {

state = "q₂";

}

else if (c.equals ("1")) {

state = "q₂";

}

break;

}

}

System.out.println ("State: " + state);

if (state.equals ("q₀")) {

Sopln ("Your String is accepted");

} else {

Sopln ("String Rejected");

WY

Output:

Enter the Binary String:

110

001

10101

01010

01101

10010

111101

000010

LAB - 7

- AIM : Design a program to find 1's complement using Mealy Machine
- import java.util.*;

```
public class Main {
    public static void main (String [] args) {
```

~~Scanner myObj = new Scanner (System.in);
System.out.println ("Enter the Binary String :");
String str = myObj.nextLine();~~
~~String state = "q1";
String result = "";
for (int i = 0; i < str.length(); i++) {
 char ch = str.charAt(i);
 String c = String.valueOf(ch);~~

switch (state) {

case "q1":

~~if (c.equals ("0")) {
 result = result + "1";
} else if (c.equals ("1"))
 result = result + "0";~~

}
break;

System.out.println (result);

LAB-8

- AIM: Design a program to find 1's complement using Moore machine.

→ import java.util.*;

```
public class Main {
    public static void main (String [] args) {
```

```
        Scanner sc = new Scanner (System.in);
        System.out.println ("Enter the Binary String :");
        String str = sc.nextLine();
```

~~String state = "q0";~~

~~String result = "";~~

```
for (int i=0; i < str.length(); i++) {
    char ch = str.charAt(i);
    String c = String.valueOf(ch);
```

~~switch (state) {~~

~~case "q0":~~

```
    if (c.equals ("0")) {
```

~~state = "q0";~~

```
    } else if (c.equals ("1")) {
```

~~state = "q1";~~

~~} break;~~

case "q₁:

```
if (c.equals ("0")) {  
    state = "q0";  
} else if (c.equals ("1")) {  
    state = "q1";  
}
```

break;

}

if (state.equals ("q₀")) {
 result = result + "1";

} else {

result = result + "0";

}

System.out.println (result);

}

10/12/2018

LAB - 9

- AIM: Write a LEX program to count no. of vowels, consonants and digit in a given string.

% d

#include <stdio.h>

int v=0, d=0, c=0;

% }

%%

[AEIOUaeiou] {v=v+1}

[0-9] {d=d+1}

[^AEIOUaeiou] {c=c+1}

%%

int yywrap()

{ return 1;

}

int main()

{

printf("Enter the string : \n");

yylex();

printf("Vowels : %d\n", v);

printf("Digits : %d\n", d);

printf("Consonants : %d\n", c);

return 0;

}

WLR/22R

LAB-10

- AIM: Write a LEX program to get the ECHO of a string.

% {

#include <stdio.h>

% }

% %

.ECHO :

ECHO :

% %

int yywrap ()

{ return 1;

int main ()

{

yylex();

return 0;

}

Up 2d

LAB-11

Page : 21
Date : 20/12/24

- AIM: write a LEX program to count and recognize the keywords and identifiers.

↳ % {

```
#include <stdio.h>
int Keywords = 0, K = 0;
```

% }

(key | adarsh | lab) automata) {Keywords = Keywords + 1;

printf ("This is a Keyword"); } }

[a-zA-Z][a-zA-Z0-9]* {K = K + 1; printf ("It is a identifier"); }

* { printf ("it is neither a keyword nor a identifier"); }

int yywrap()

{

return 1;

int main()

{

printf ("Enter the string : \n");

yylex();

return 0;

WVZ

LAB-12

Page : 22
Date : 20/2/24

- ① AIM: Write a LEX program to find words beginning and ending with 'a'.

↳ %.

```
#include <stdio.h>
int K=0;
```

%%

```
[a][a-zA-Z0-9]*[a] {K=K+1; printf("This starts and ends
with a");}
```

* {printf("This doesn't start or end with a");}

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

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

What's the

LAB-13

Page : 23
Date : 27/02/24

- AIM: Write a LEX program to identify the capital letters, small letters, digits and special symbols.

→ % {

include <stdio.h>

int c=0, s=0, d=0, sy=0 ;

% }

%%

[a-z] { s = s + 1 ; printf ("small letter"); }

[A-Z] { c = c + 1 ; printf ("capital letter"); }

[0-9] { d = d + 1 ; printf ("digit"); }

* { sy = sy + 1 ; printf ("symbol"); }

%%
int yywrap()

return 1;

}

int main ()

{

printf ("Enter the string : \n");

yylex();

printf ("capital letters : %d \n", c);

printf ("small letter : %d \n", s);

printf ("digit : %d \n", d);

printf ("symbol : %d \n", sy);

return 0;

}

LAB-14

Page : 24
Date : 27/02/23

- AIM: Write a LEX program to check the valid email id.

↳ % {

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

% }

%%

```
[a-zA-Z][a-zA-Z.0-9]* + @ [a-zA-Z]+\.[a-zA-Z] { printf ("valid  
email!\n"); }  
.* { printf ("invalid email\n"); }  
%
```

~~int yyparse () {
return 1;
}~~

```
int main () {  
printf ("Enter input\n");  
yylex ();  
return 0;  
}
```

verified

LAB - 15

Page : 25
Date : 27/02/24

- ① AIM: Write a LEX program to check the valid mobile number.

↳ % {

#include <stdio.h>

% }

% %

[6-9]{1}[0-9]{5} {printf("a valid mobile no."); }
.* {printf("not a valid mobile no."); }
%

int yywrap()
{

return 1;
}

int main()

{
printf("Enter mobile no. : \n");

yylex();

return 0;
}

LAB-16

- Aim: Design PDA for grammar $a^n b^n$ for $n > 1$

```

import java.util.Scanner;
import java.util.Stack;
public class pda {
    public static void main (String [] args) {
        Scanner sc = new Scanner (System.in);
        System.out.println ("Enter");
        String str = sc.nextLine ();
        Stack<Character> stack = new Stack<> ();
        String state = "q0";
        int i = 0;
        while (i < str.length ()) {
            if (state.equals ("q0") && str.charAt (i) == 'a') {
                stack.push (str.charAt (i));
                i++;
            } else if (state.equals ("q1") && str.charAt (i) == 'b') {
                if (stack.isEmpty ()) {
                    break;
                }
                stack.pop ();
                i++;
            } else {
                break;
            }
        }
        if (i == str.length () && stack.isEmpty ()) {
            System.out.println ("String Valid");
        } else {
            System.out.println ("String Invalid");
        }
    }
}

```

LAB - 17

- AIM : Design a turing machine for $m + m$ where m and n are unary.

import java.util.*;

public class Main {

 public static void main (String [] args) {

 char [] arr = new char [25];

 Arrays.fill (arr, 'A');

 Scanner sc = new Scanner (System.in);

 String s = scanner.next();

 int j = 0;

 for (int i = 5; i < s.length(); i++) {

 arr [i] = s.charAt (j++);

}

 int index = 5;

 String state = "q0";

 while (true) {

 if (state.equals ("q0") && arr [index] == '1') {

 arr [index] = 'A';

 index++;

 state = "q1";

}

 else if (state.equals ("q1") && arr [index] == '1') {

 index++;

 else if (state.equals ("q1") && arr [index] == 'c') {

 state = "q2";

 index++;

```
else if (state.equals("q4") && arr[index] == 'A') {  
    state = "q0";  
    index++;  
}  
else if (state.equals("q0") && arr[index] == 'C') {  
    arr[index] = 'A';  
    state = "qf";  
    index++;  
}  
else {  
    break;  
}  
for (int i = 0; i < arr.length; i++) {  
    System.out.print(arr[i]);  
}  
System.out.println();
```

up 18/4