Practical -
Gr. No. - A3

Page No.	
Class	
Roll No.	
Date :	/ /201

Title: Write C program to recognize string under 'a*', 'a*b', 'abb'.

Theory :-

A regular Expression can be recursively defined as follows.

$(L(E) = \{E\})$

ϕ is a regular expression denoting an empty language

$(L(\phi)) \{\}$

- x is a regular expression when
$L = \{x\}$

- If x is regular expression denoting the language $L(x)$ & y is a regular expression denoting the language $L(Y)$ then,

- $x+y$ is a regular expression corresponding to the language $L(x)$ & y is a regular expression denoting the language $L(Y)$ then.

- x+y is a regular expression corresponding to the language L(x). ~~& y is a regular expression~~ L(y) where L(x+y) = L(x) + L(y)

- x*y is a regular expression corresponding to the language L(x). L(y) where L(x,y) = L(x). ∈ L(y)

R^* is a regular expression corresponding to the language.

If we apply any of the rules several times from 1 to 5, they are regular expression.

Regular expression can be defined by the following rules.

1. Every letter of the alphabet ∑ is a regular expression.

2. NULL string ∈ & empty set ϕ are regular expressions.

3. If r_1 & r_2 are regular expressions, then
   (i) r_1, r_2
   (ii) $r_1 r_2$ (concatenation of r_1, r_2)

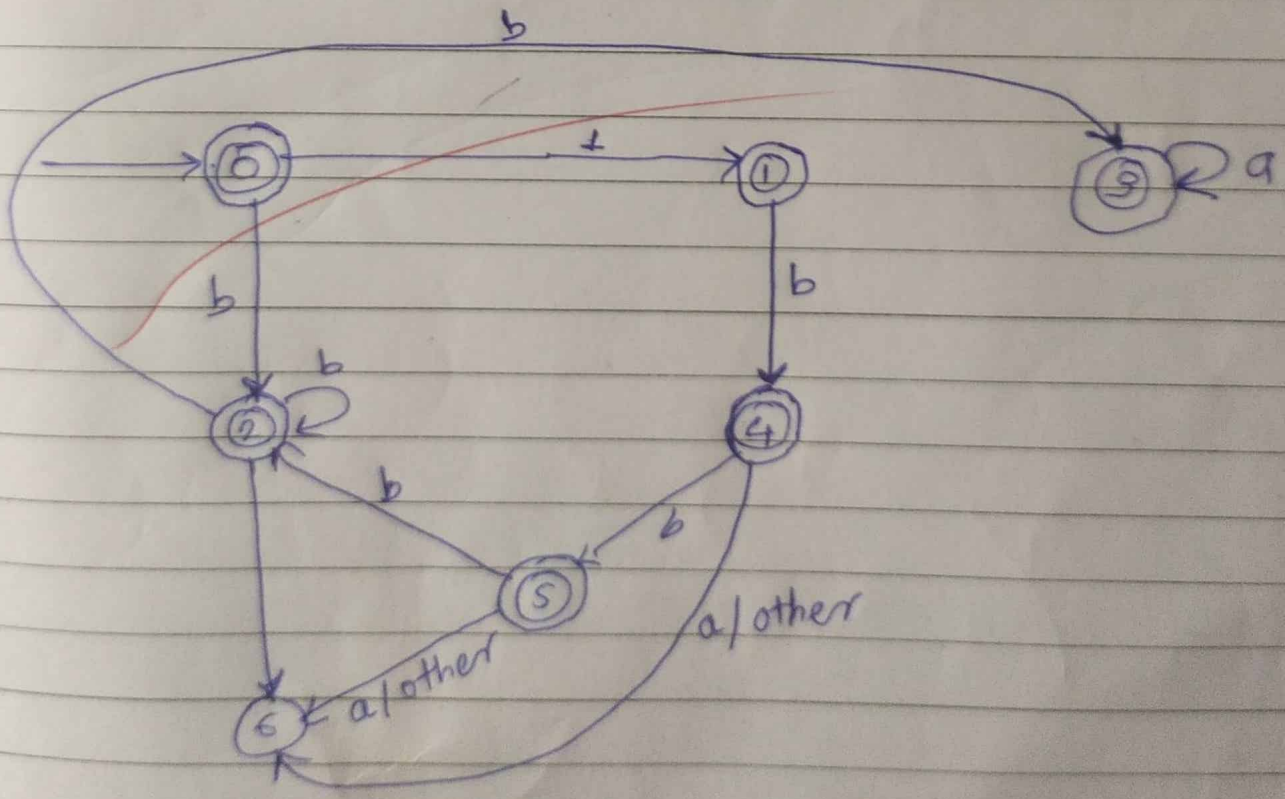
(iii) $r_1 + r_2$ (union of r_1 & r_2)

(iv) r_1^*, r_2^* (Kleen clausure of r_1 & r_2) are also regular expression.

(4) If a sharing can be derived from the rules 1, 2 & 3 then it is also a regular expression.

Some RE example:

Regular expression	Regular set
$(a + 1.0^*)$	$L = \{0, 1, 10, 100, 1000, \cdots\}$
$(0^* . 0^*)$	$L = \{1, 01, 10, 010, 0010, \cdots\}$
$(a+\epsilon)(1+\epsilon)$	$L = \{\epsilon, 0, 1, 0, 1\}$



DFA of regular expression a^*, a^*b, abb

Page No. 14
Class
Roll No.
Date : / /201

Result:

Thus the program to recognize string under 'a*', 'a*b', 'abb' was executed and the output was verified successfully.

```
else
printf("\n String not accepted by automata.");
}
```

Output :

```
PS C:\c programs> gcc A3.cpp

PS C:\c programs> .\a.exe

Enter a string:aaaaa

aaaaa is accepted under rule 'a*'

PS C:\c programs> .\a.exe

Enter a string:abbbbb

abbbbb is accepted under rule 'a*b+'

PS C:\c programs> .\a.exe
```