

Outputs

Lab 1

| PROBLEMS | OUTPUT | DEBUG CONSOLE | TERMINAL | PORTS |
|--|--------|---------------|----------|-------|
| PS E:\Sarfraz\4th SEMESTER\Theory of Computation> cd "e:\Sarfraz\4th SEMESTER\Theory of Computation" ; if (\$?) { .\Lab12 } ● Accepted | | | | |

Lab 2

| PROBLEMS | OUTPUT | DEBUG CONSOLE | TERMINAL | PORTS |
|--|--------|---------------|----------|-------|
| PS E:\Sarfraz\4th SEMESTER\Theory of Computation> cd "e:\Sarfraz\4th SEMESTER\Theory of Computation" ; if (\$?) { .\Lab13 } ● ACCEPTED | | | | |

Lab 3

| PROBLEMS | OUTPUT | DEBUG CONSOLE | TERMINAL | PORTS |
|--|--------|---------------|----------|-------|
| PS E:\Sarfraz\4th SEMESTER\Theory of Computation> cd "e:\Sarfraz\4th SEMESTER\Theory of Computation" ; if (\$?) { .\Lab14 } ● Enter the string to be checked: 01 String is accepted | | | | |

Lab 4

| PROBLEMS | OUTPUT | DEBUG CONSOLE | TERMINAL | PORTS |
|--|--------|---------------|----------|-------|
| PS E:\Sarfraz\4th SEMESTER\Theory of Computation> cd "e:\Sarfraz\4th SEMESTER\Theory of Computation" ; if (\$?) { .\Lab15 } ● Enter a string of 0s and 1s: 0011 Input rejected ● PS E:\Sarfraz\4th SEMESTER\Theory of Computation> cd "e:\Sarfraz\4th SEMESTER\Theory of Computation" ; if (\$?) { .\Lab15 } Enter a string of 0s and 1s: 01 Input accepted | | | | |

Lab 5

```
PS E:\Sarfraz\4th SEMESTER\Theory of Computation> cd "e
; if ($?) { .\Lab16 }
● Enter a string of 0s and 1s: 110
Input accepted
● PS E:\Sarfraz\4th SEMESTER\Theory of Computation> cd "e
; if ($?) { .\Lab16 }
Enter a string of 0s and 1s: 0
Input accepted
```

Lab 6

```
● PS E:\Sarfraz\4th SEMESTER\Theory of Computation> cd "e:\Sarf
; if ($?) { .\Lab17 }
Enter number of nodes: 3
edge node ...):
1 0 2 0 1 1 2
2 0 2 0 2 1 3
3 1 1 0 3

===== INPUT DONE =====

Accepted strings by NFA:
11
011
101
110
0011
0101
0110
1001
1010
1100
```

Lab 7

```
● PS E:\Sarfraz\4th SEMESTER\Theory of Computation> cd "e
; if ($?) { .\Lab18 }
Building NFA for regex a*
Enter the string to test: aaa
String Is Accepted
```

Lab 8PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS

- PS E:\Sarfraz\4th SEMESTER\Theory of Computation> **cd "e:\Sarfraz\4th SEMESTER\Theory of Computation"**

```
Follow zero-based indexing for states.
Enter the number of states: 2
Enter number of final states: 1
Enter final states: 1
Enter number of NFA transitions: 4
Enter transitions as: initial_state input_symbol final_state
0 0 0
0 1 1
1 0 0
1 1 1
Enter initial state: 0

Solving according to DFA

The total number of distinct DFA states are:
STATE  0  1
q0 1 2
q1 1 2

Enter string to test (0 and 1 only): 11
String path: 1-2-2-
Final state: 2
String Accepted
```