1914078 Devansh
B1 Tutorial

Q1) ii) Method 1 : Equivalence theorem

Step 1 : Drawing a state transition table

	a	b
0	0	0
→ 1	2	3
2	4	5
3	2	6
4	2	3
5	0	7
6	2	6
7	8	5
8	8	5

Step 2 : Now using equivalence theorem, we will separate final and non final states in sets.

$\therefore P_0 = \{\{1,2,3,4,6,7,8\}, \{0,5\}\}$

(final) (non-final)

$P_1 = \{\{1,3,4,6\}, \{2,7,8\}, \{5\}, \{0\}\}$

$P_2 = \{\{1,3,4,6\}, \{2\}, \{7,8\}, \{5\}, \{0\}\}$

$P_3 = \{\{1,3,4,6\}, \{2\}, \{7,8\}, \{5\}, \{0\}\}$

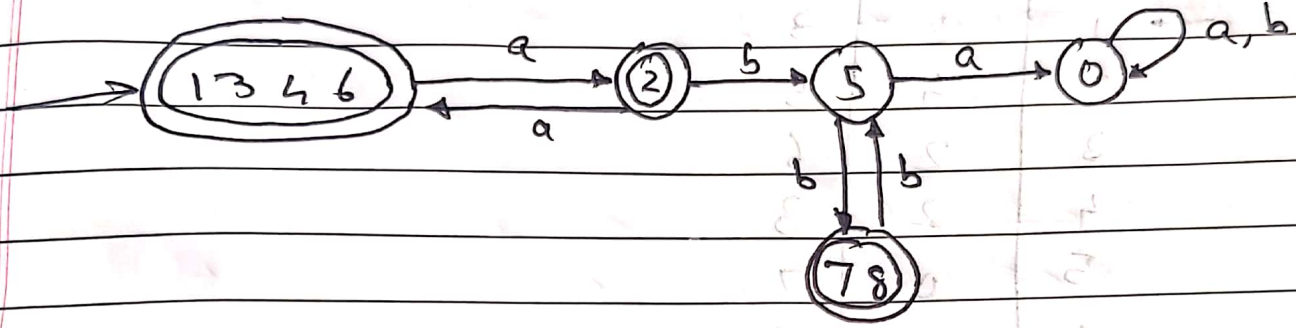
↳ minimal DFA

Scanned with CamScanner

∴ $P_2 = P_3$, we stop

From P_3, we infer that states {1, 3, 4, 6} and states {7, 8} are equivalent and thus can be merged together to create their respective separate state

The minimal DFA is :-



Scanned with CamScanner

Method 2: Using table filling method

Step 1: Draw table of all states pairing

	1	2	3	4	5	6	7	8	0
1	X	X	X	X	X	X	X	X	X
2	✓	X	X	X	X	X	X	X	X
3		✓	X	X	X	X	X	X	X
4		✓		X	X	X	X	X	X
5	✓	✓	✓	✓	X	X	X	X	X
6		✓			✓	X	X	X	X
7	✓	✓	✓	✓	✓	✓	X	X	X
8	✓	✓	✓	✓	✓	✓		X	X
0	✓	✓	✓	✓	✓	✓	✓	✓	X

We cross out the upper Δ so that we have unique states

Step 2: Now we will mark all pairs where P ∈ final state (F)

For unmarked pair (P, Q) such that [δ(P, x), δ(Q, x)] is marked then mark [P, Q]. For any 'x' being in input symbol

for 2, 1

2, a → 4

1, a → 2

Here we know that 5, 0 are the only non final states, we need a pair of (P, Q) such that P, Q ≠ 0, 5 or 5, 0

Eg) either P or Q is 5, 0

Scanned with CamScanner

So, for pairs:

1,2 $\xrightarrow{a}$ 2,4 ✗
$\xrightarrow{b}$ 3,5 ✓

Such nodes will be marked ✓ is the same table

1,3 $\xrightarrow{a}$ 2,2 ✗
$\xrightarrow{b}$ 3,6 ✗

2,3 $\xrightarrow{a}$ 4,2 ✗
$\xrightarrow{b}$ 5,6 ✓

1,4 $\xrightarrow{a}$ 2,2 ✗
$\xrightarrow{b}$ 3,3 ✗

2,4 $\xrightarrow{a}$ 4,2 ✗
$\xrightarrow{b}$ 5,3 ✓

3,4 $\xrightarrow{a}$ 2,2 ✗
$\xrightarrow{b}$ 6,3 ✗

1,6 $\xrightarrow{a}$ ✗
$\xrightarrow{b}$ ✗

2,6 $\xrightarrow{a}$ ✗
$\xrightarrow{b}$ 5,6 ✓

3,6 $\xrightarrow{a}$ ✗
$\xrightarrow{b}$ ✗

4,6 $\xrightarrow{a}$ ✗
$\xrightarrow{b}$ ✗

1,7 $\xrightarrow{a}$ ✗
$\xrightarrow{b}$ 3,5 ✓

2,7 $\xrightarrow{a}$ ✗
$\xrightarrow{b}$ 5,5 ✗

3,7 $\xrightarrow{a}$ ✗
$\xrightarrow{b}$ 6,5 ✓

4,7 $\xrightarrow{a}$ ✗
$\xrightarrow{b}$ 3,5 ✓

6,7 $\xrightarrow{a}$ ✗
$\xrightarrow{b}$ 6,5

1,8 $\xrightarrow{a}$ ✗
$\xrightarrow{b}$ 5,3 ✓

2,8 $\xrightarrow{a}$ ✗
$\xrightarrow{b}$ 5,5, ✗

3,8 $\xrightarrow{a}$ ✗
$\xrightarrow{b}$ 6,5 ✓

4,8 $\xrightarrow{a}$ ✗
$\xrightarrow{b}$ 3,5 ✓

6,8 $\xrightarrow{a}$ ✗
$\xrightarrow{b}$ 6,5, ✓

7,8 $\xrightarrow{a}$ ✗
7,8 $\xrightarrow{b}$ 5,5 ✗

0,5 $\xrightarrow{a}$ 0,0 ✗
$\xrightarrow{b}$ 0,7 ✓

Step 4: Repeat step iii until no new markings
[(4,8), (8,2)]

1,3 $\xrightarrow{a}$ 2,2 ✗
$\xrightarrow{b}$ 3,6 ✗

1,4 $\xrightarrow{a}$ 2,2 ✗
$\xrightarrow{b}$ 3,3 ✗

3,4 $\xrightarrow{a}$ 2,2 ✗
$\xrightarrow{b}$ 6,3 ✗

1,6 $\xrightarrow{a}$ 2,2 ✗
$\xrightarrow{b}$ 3,6 ✗

3,6 $\xrightarrow{a}$ 2,2 ✗
$\xrightarrow{b}$ 6,6 ✗

4,6 $\xrightarrow{a}$ 2,2
$\xrightarrow{b}$ 3,6 ✗

7,2 $\xrightarrow{a}$ 4,8 ✓

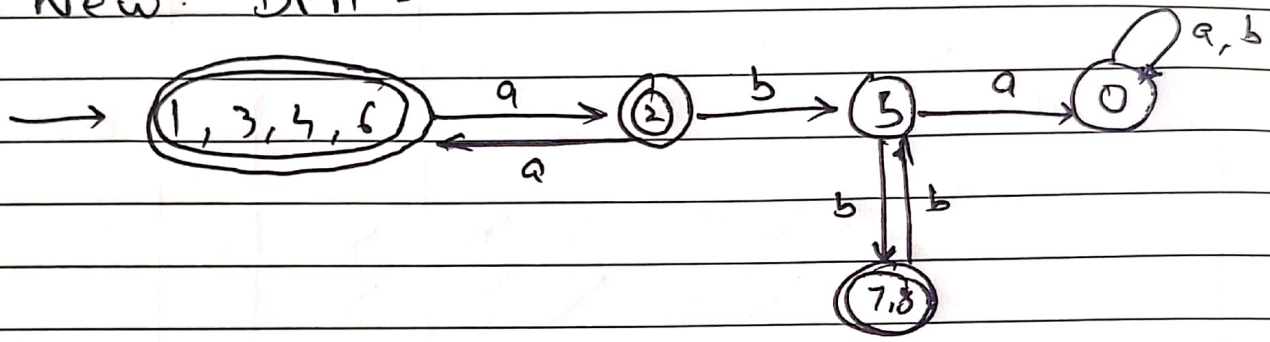
8,2 $\xrightarrow{a}$ 8,4

Scanned with CamScanner

No more recursions possible

v) Combining unmarked states
Unmarked States → (3,1), (4,1), (4,3), (6,1), (6,3), (6,4), (7,8)

vi) New DFA :-



Scanned with CamScanner

Q2) The purpose of minimizing states in DFA are:-

(i) To reduce the amount of space required to store DFA

(ii) To reduce the complexity of understanding how it works

(iii) To increase the performance speed of the program

(iv) Minimizing an expression involves merging two or more states into a single equivalent state. This should produce a smaller automation that accomplishes the same tasks.

Scanned with CamScanner