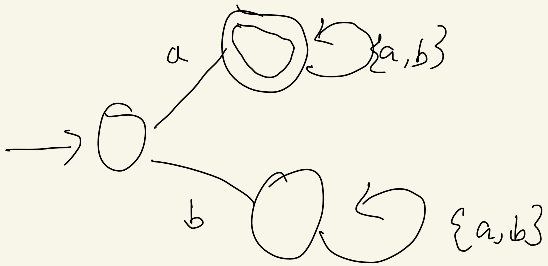
1.4

e) L is combination of 2 languese, L_1 and L_2

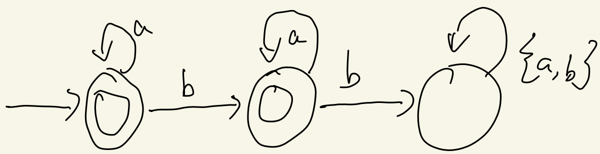
L_1 is $\{w \mid w \text{ starts with } a\}$

L_2 is $\{w \mid w \text{ has at most } 1\ b\}$

The DFA for L_1 is



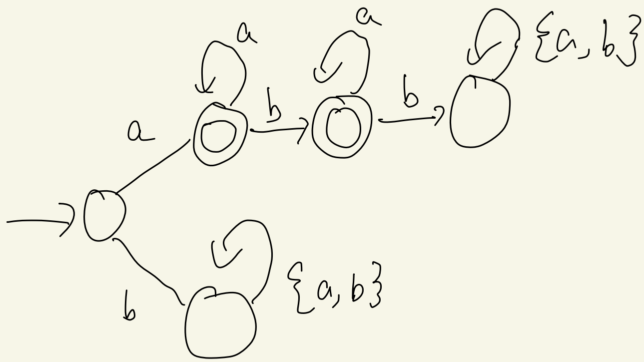
The DFA for L_2 is:



L is the intersection of L_1 and L_2

So M regconize L is and only if it recognize both L_1 and L_2.

$\Sigma = \{a, b\}$



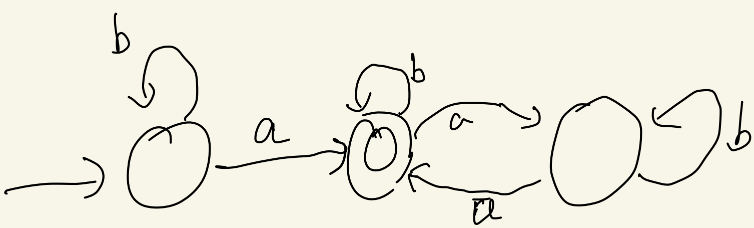
f)

L is combination of 2 languages, L_1 and L_2

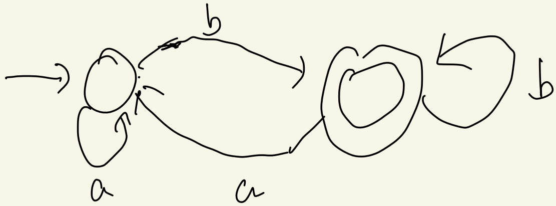
L_1 is $\{w \mid w$ has odd number of a's$\}$

L_2 is $\{w \mid w$ ends with b$\}$

The DFA for L_1 is



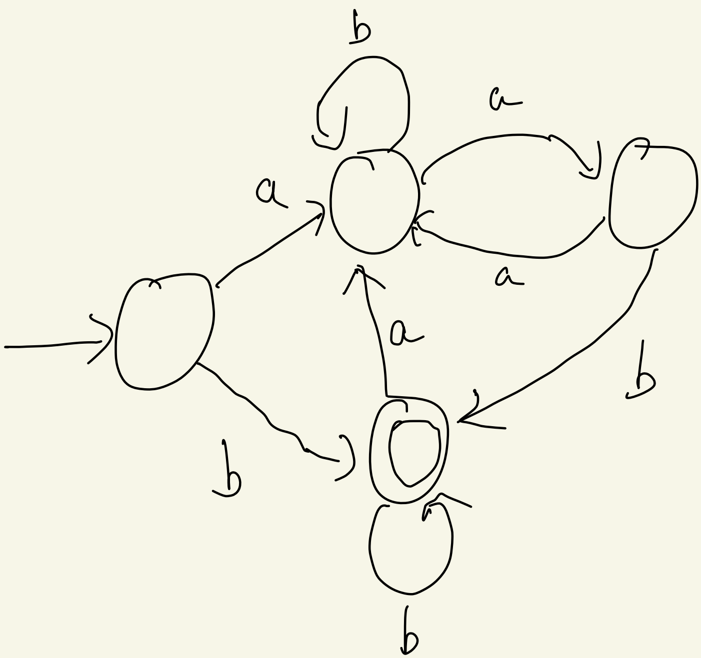
The DFA for L_2 is:



L is the intersection of L_1 and L_2

So M regcognize L is and only if it recognize both L_1 and L_2.

$\Sigma = \{a, b\}$



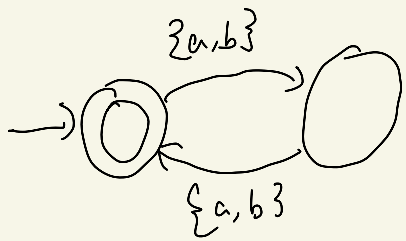
9)

L is combination of 2 languages, L_1 and L_2

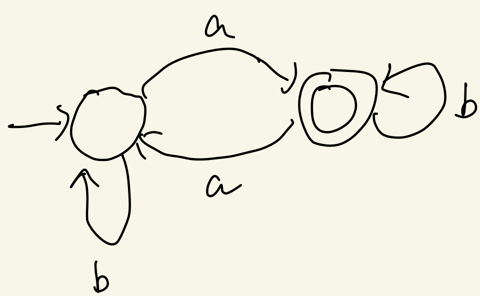
L_1 is $\{w \mid w$ has even length$\}$

L_2 is $\{w \mid w$ has odd number of a's$\}$

The DFA for L_1 is:



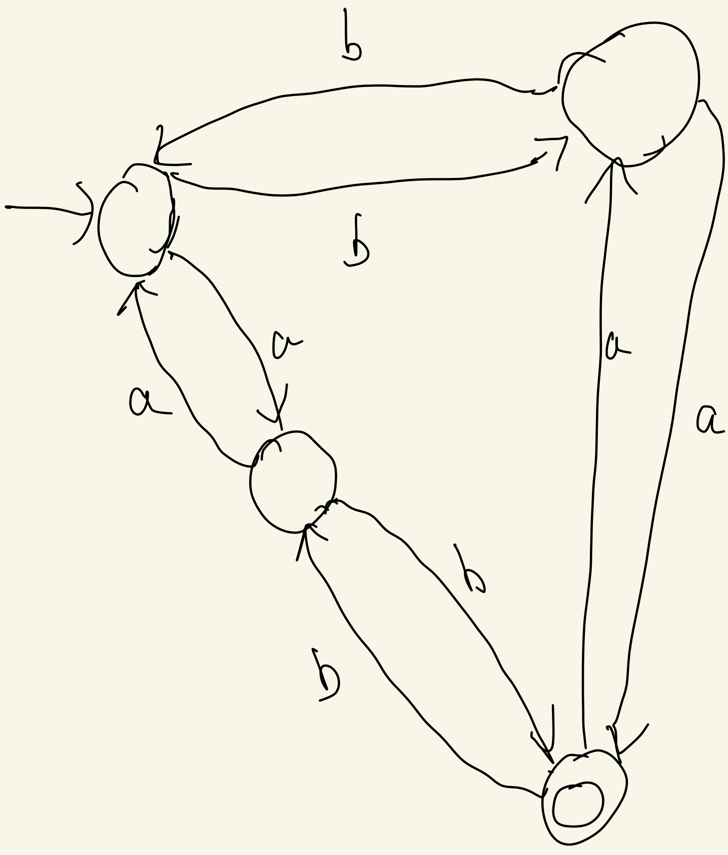
The DFA for L_2 is:



L is the intersection of L_1 and L_2

So M regcognize L is and only if it recognize both L_1 and L_2.

$\delta = \{a, b\}$



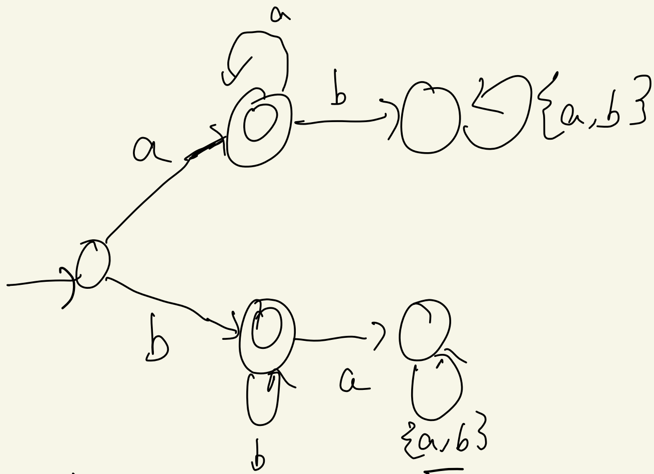
1.5

c) L is $\{w \mid w$ contains neither ab nor ba$\}$

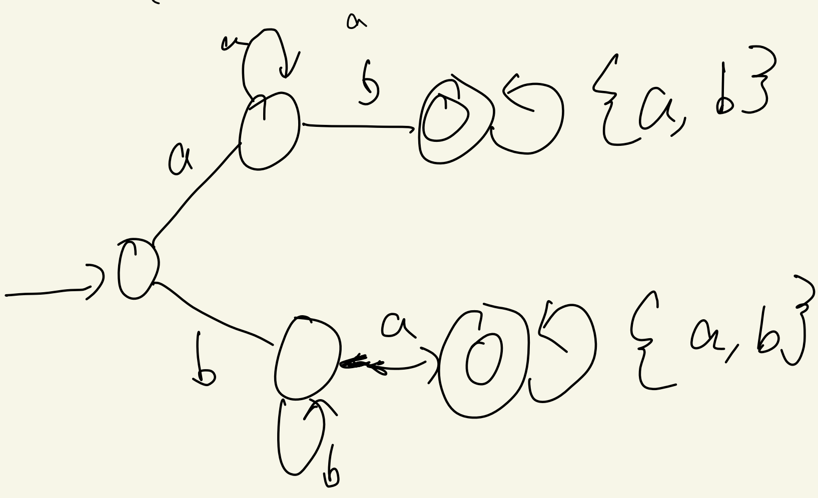
So the complement of L is

$\bar{L} = \{w \mid w$ contains either ab or ba$\}$

The DFA for L is



The DFA for $\bar{L}$ is



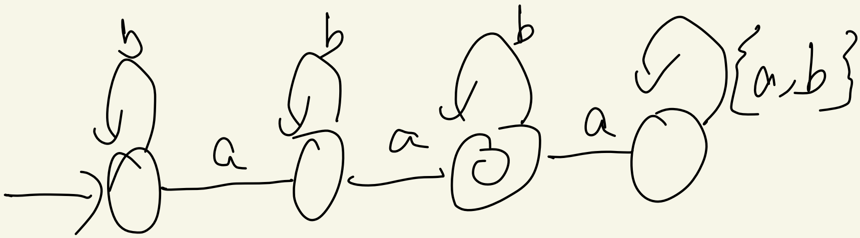
g)

L is {w | w is any string that doesn't have exactly 2 a's}

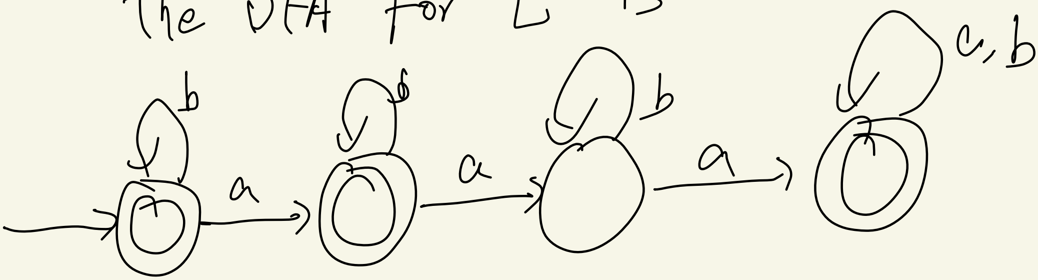
So the complement of L is

$\bar{L}$ = {w | w is any string that has exactly 2 a's}

The DFA for L is



The DFA For $\bar{L}$ is



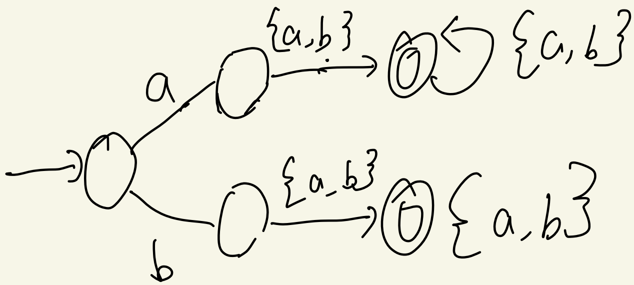
h)

L is $\{w \mid w$ is any string except a and b$\}$

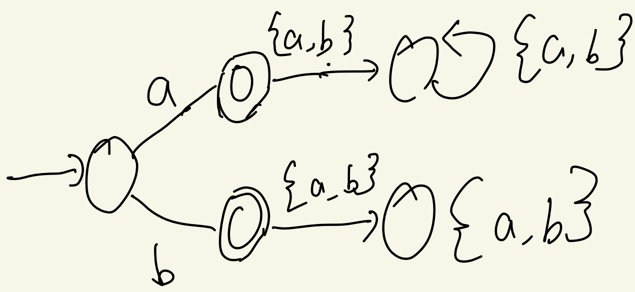
So the complement of L is

$\bar{L} = \{w \mid w$ is exactly a or b$\}$

The DFA for L is

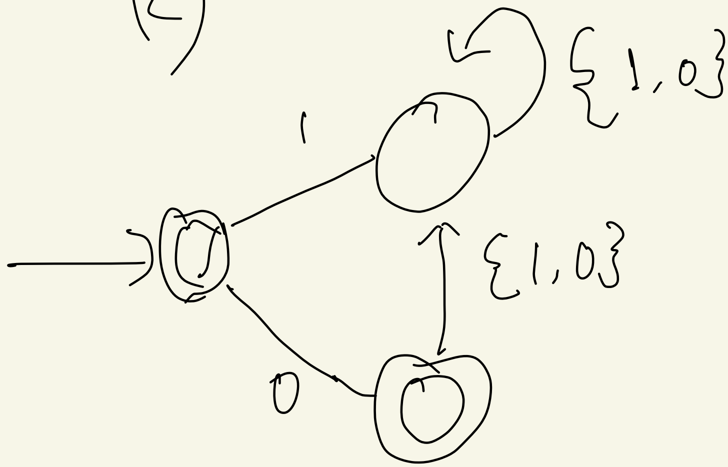


The DFA for $\bar{L}$ is

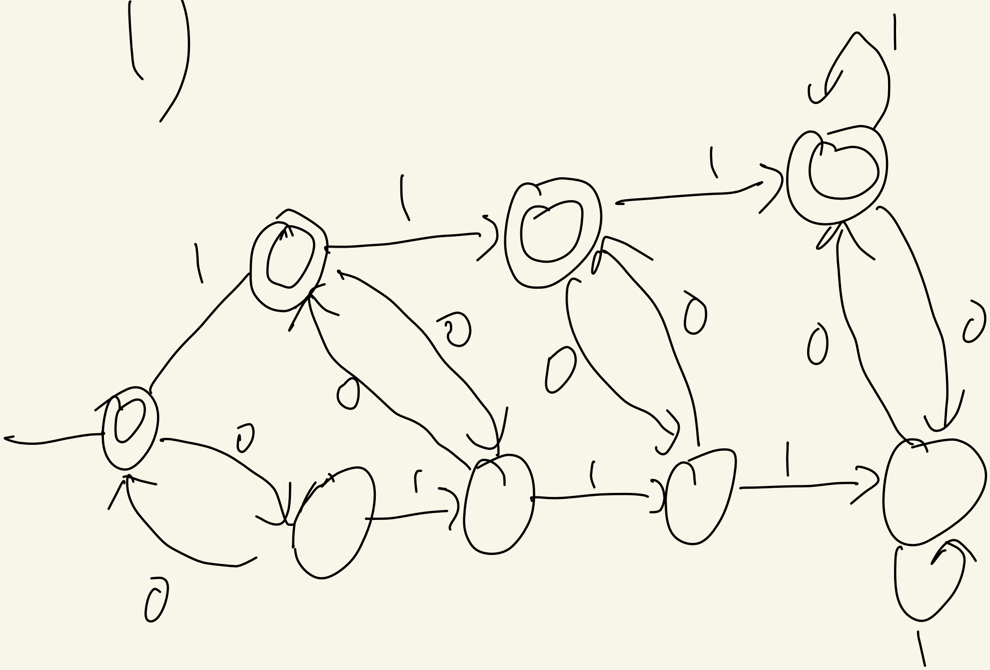


1.6

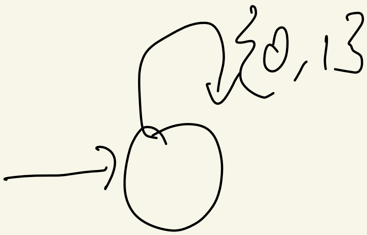
(k)



l)

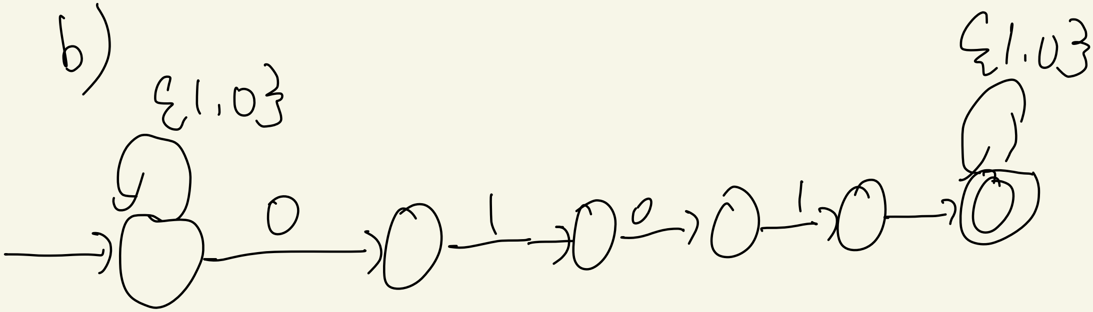


m)



1.7

b)

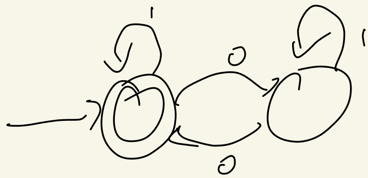


c) L is a combination of 2 sub-language L_1, L_2

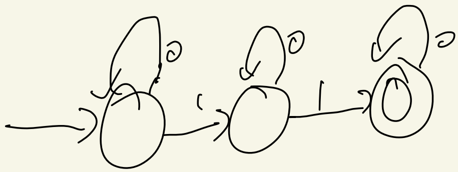
$L_1 = \{w \mid w$ contains even number of 0s$\}$

$L_2 = \{w \mid w$ contains exactly 2 1s$\}$

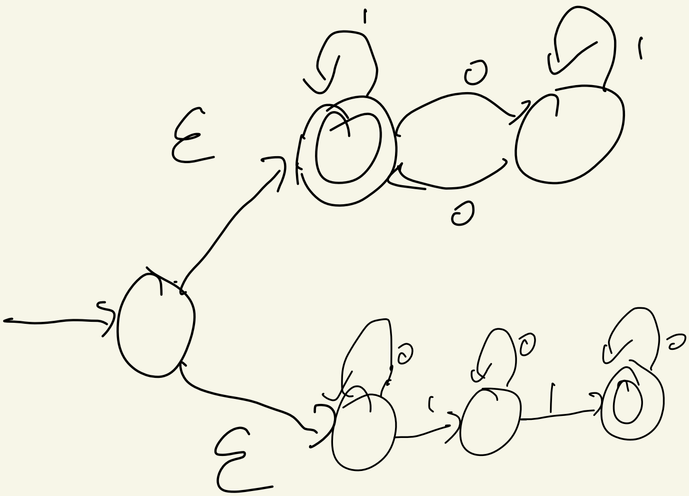
The DFA For L_1 is:



The DFA for L_2 is:



To find L, we just combine L_1, L_2 together into a NFA by using ϵ



1.14

a) Imagine we have a machine M that accepts Language A

We build another machine $\overline{M}$ from M with accept states and non-accept states swapped.

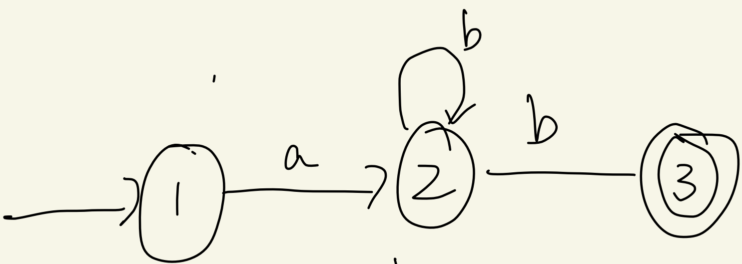
$\overline{M}$ will not accept any of language A while M accept it.

So $\overline{M}$ accept language that is not accepted by M.

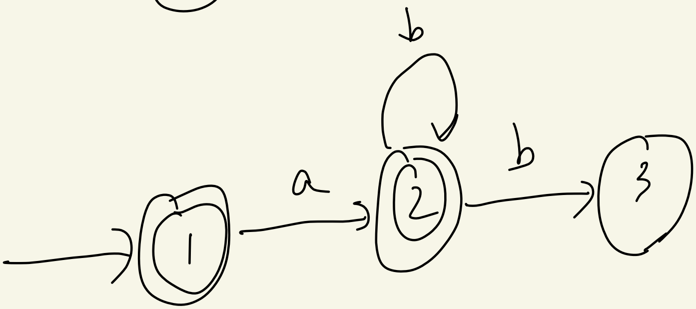
$\overline{M}$ recognize language that is Complemented to the language A.

Therefore class of regular language is closed under complement.

b) Because NFA promises multiple different "future" state. The class of languages recognized by NFA is not closed under complement.



M



$\bar{M}$

string "abb" is accepted by both machines. Therefore, by swapping accept states and non-accept state doesn't always yeild the complement. And class of language recognized by NFA is not closed under complement.

1.16

a) Following Theorem 1.39, to construct DFA from NFA, we just need to build the $(Q', \Sigma', \delta', q_0', F)$ from DFA

$Q' = P(Q) = \{\emptyset, 1, 2, (1, 2)\}$

$\Sigma' = \Sigma = \{a, b\}$

$\delta'(R, a) = \{q \in Q \mid q \in \delta(r, a) \text{ for } r \text{ in } R\}$

$\delta'(\emptyset, a) = \emptyset \quad \delta'(\emptyset, b) = \emptyset$

$\delta'(\{1\}, a) = \delta(1, a) = \{1, 2\}$

$\delta'(\{1\}, b) = \delta(1, b) = \{2\}$

$$\delta'(\{2\}, a) = \delta(2, a) = \phi$$

$$\delta'(\{2\}, b) = \delta(2, b) = \{1\}$$

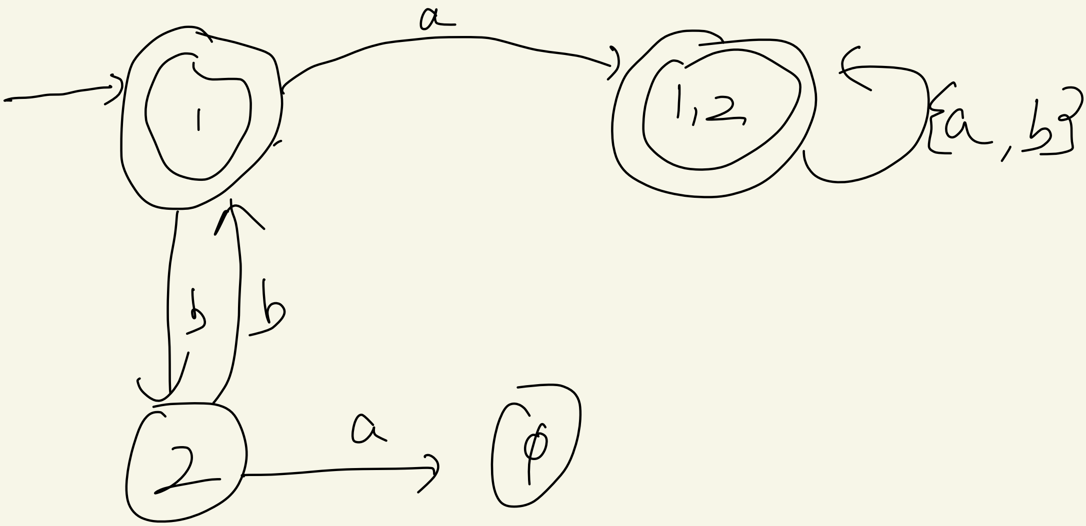
$$\delta'(\{1,2\}, a) = \delta(1, a) \cup \delta(2, a)$$
$$= \{1, 2\}$$

$$\delta'(\{1,2\}, b) = \delta(1, b) \cup \delta(2, b)$$
$$= \{1, 2\}$$

$$q' = \{q_0\} = \{1\}$$

F' = All the states containing F

$$= (\{1\}, \{1, 2\})$$



b)

Following Theorem 1.39, to constructe DFA from NFA, we just need to build the $(Q', \Sigma', \delta', q_0', F)$ from DFA

$$Q' = P(Q) = (\emptyset, \{1\}, \{2\}\ \{3\}, \{1,2\}, \{2,3\}, \{1,3\}, \{1,2,3\})$$

$$\Sigma' = \Sigma = \{a, b\}$$

$$\delta'(\emptyset, a) = \delta'(\emptyset, b) = \phi$$

$$\delta'(\{1\}, a) = \delta(1, a) = \{3\}$$

$$\delta'(\{1\}, b) = \delta(1, b) = \phi$$

$$\delta'(\{2\}, a) = \delta(2, a) = \{1, 2\}$$

$$\delta'(\{2\}, b) = \delta(2, b) = \phi$$

$$\delta'(\{3\}, a) = \delta(3, a) = \{2\}$$

$$\delta'(\{3\}, b) = \delta(3, b) = \{2, 3\}$$

$$\delta'(\{1,2\},a) = \delta(1,a) \cup \delta(2,a)$$
$$= \{1,2,3\}$$

$$\delta'(\{1,2\},b) = \delta(1,b) \cup \delta(2,b)$$
$$= \phi$$

$$\delta'(\{2,3\},a) = \delta(2,a) \cup \delta(3,a)$$
$$= \{1,2\}$$

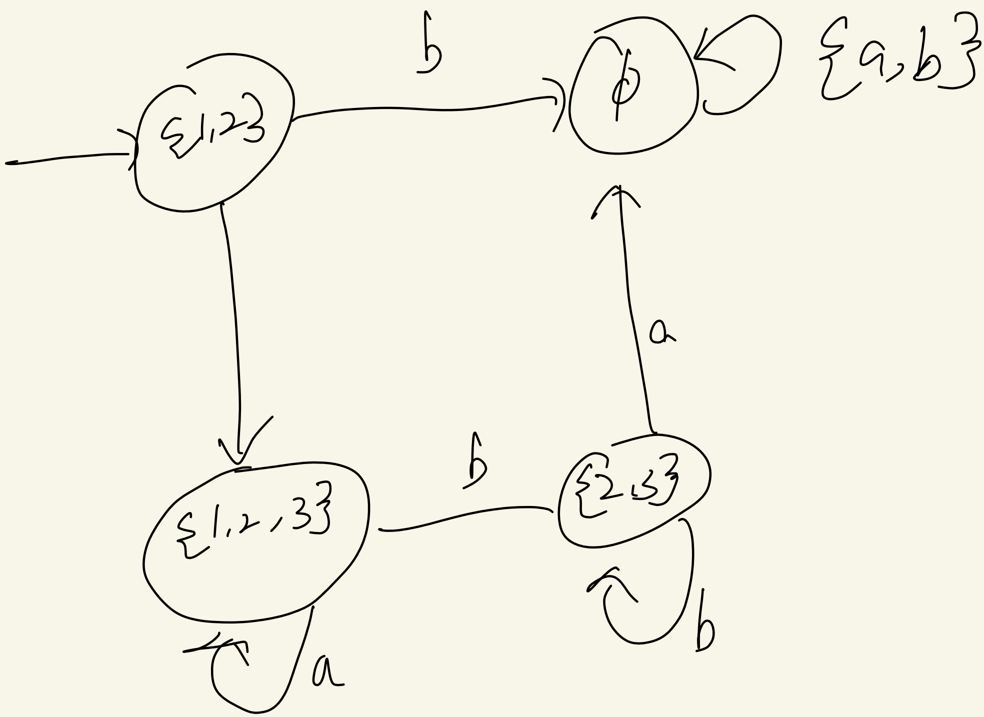
$$\delta'(\{2,3\},b) = \delta(2,b) \cup \delta(3,b)$$
$$= \{2,3\}$$

$$\delta'(\{1,3\},a) = \delta(1,a) \cup \delta(3,a)$$
$$= \{2,3\}$$

$$\delta'(\{1,3\},b) = \delta(1,b) \cup \delta(3,b)$$
$$= \{2,3\}$$

$q_0 = E(\{1\}) = \{1,2\}$

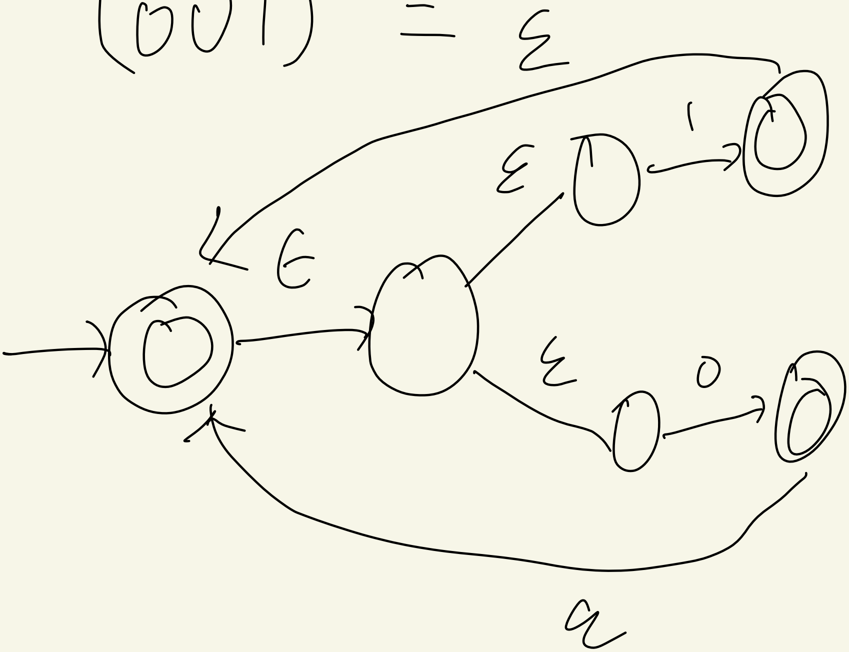
$F' =$ All the states containing 2



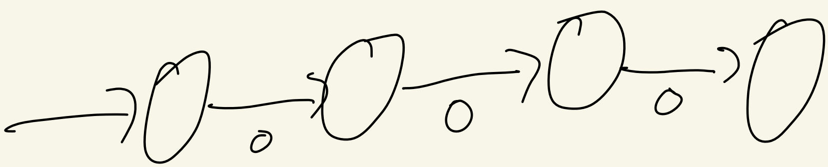
1.19

a)

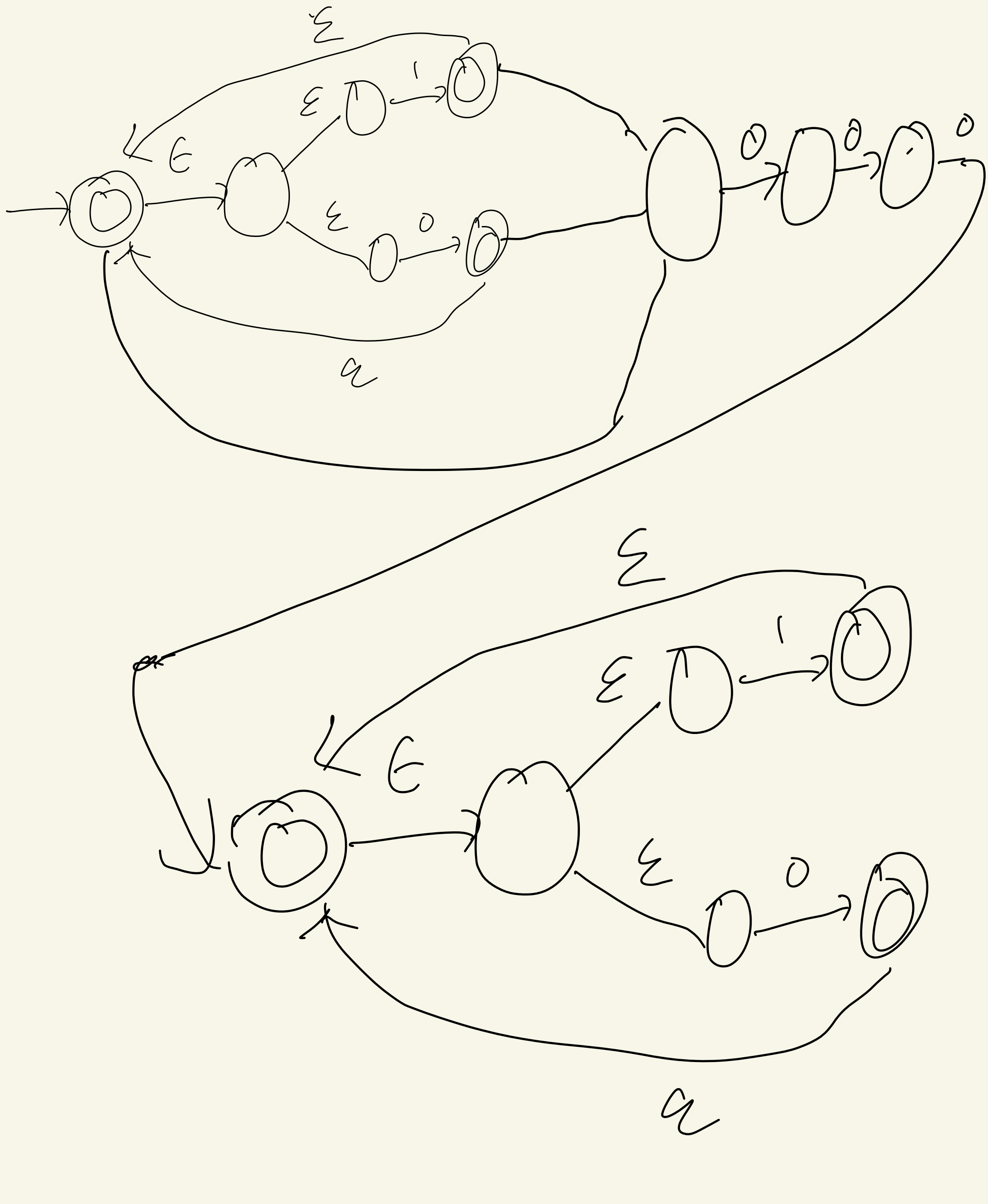
$(0\cup 1)^* =$



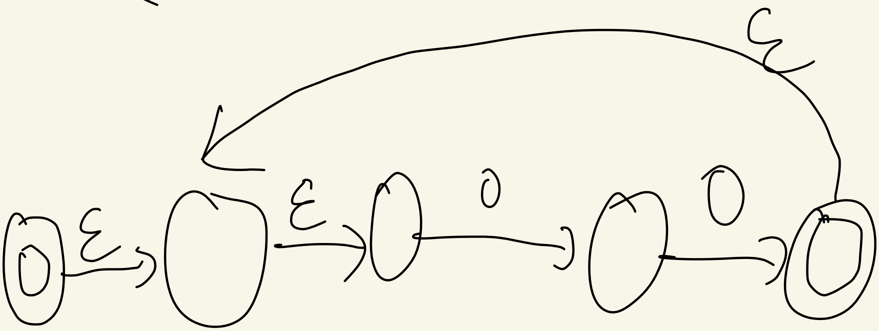
000 is



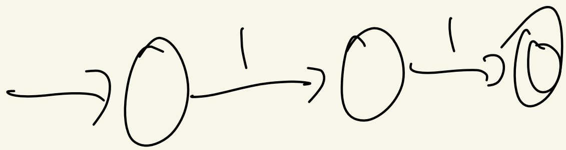
$(0 \cup 1)^* 000 (0 \cup 1)^*$ is



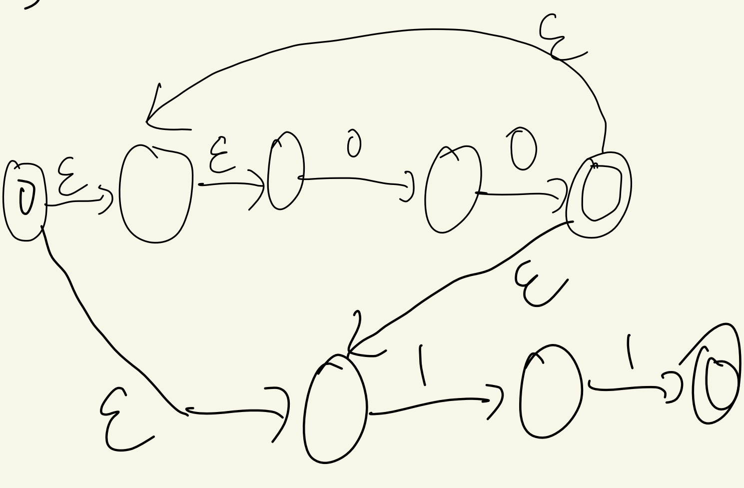
b) $(00)^*$ is



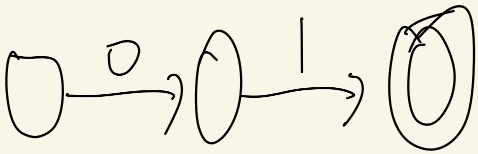
11 is



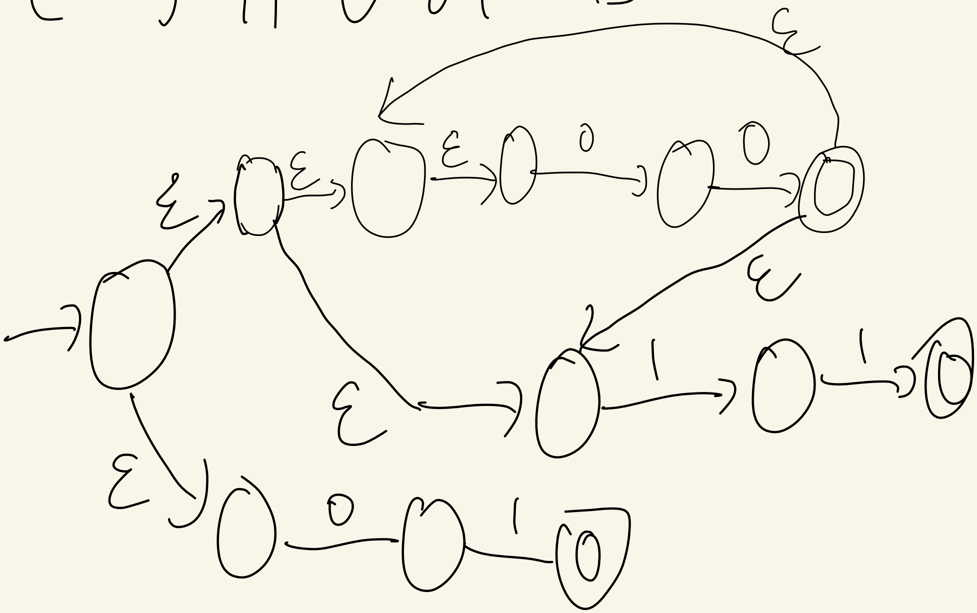
$(00)^*11$ is



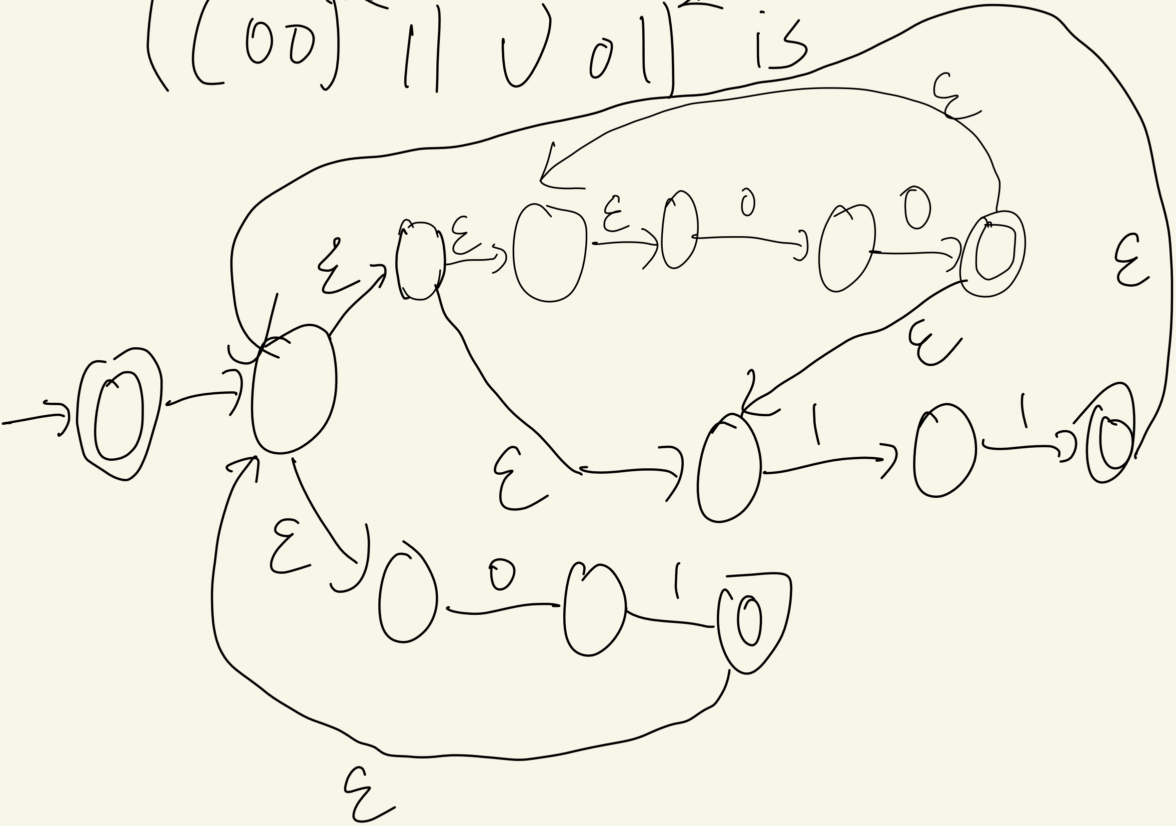
01 is



$(00)^* 11 \cup 01$ is



$((00)^* 11 \cup 01)^*$ is



c) The RE for empty set is just a accepting node.