

# Student Information

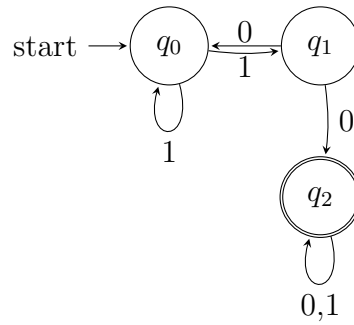
Full Name : Onur Can TIRTIR

Id Number : 2099380

## Answer 1

a.

Here is our automata:

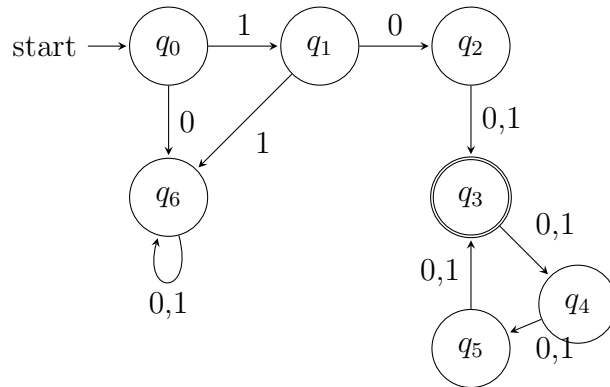


Then the transitions for the string 1011011 in our automata is as below:

$$\begin{aligned} (q_0, 11011011) &\vdash_M (q_0, 1011011) \\ &\vdash_M (q_1, 011011) \\ &\vdash_M (q_0, 11011) \\ &\vdash_M (q_0, 1011) \\ &\vdash_M (q_1, 011) \\ &\vdash_M (q_2, 11) \\ &\vdash_M (q_2, 1) \\ &\vdash_M (q_2, e) \end{aligned}$$

Then the string given above belongs to  $L_1$ .

b.



Then the transitions for the string 1000111 in our automata is as below:

$$\begin{aligned}
 (q_0, 1000111) &\vdash_M (q_1, 000111) \\
 &\vdash_M (q_2, 00111) \\
 &\vdash_M (q_3, 0111) \\
 &\vdash_M (q_4, 111) \\
 &\vdash_M (q_5, 11) \\
 &\vdash_M (q_3, 1) \\
 &\vdash_M (q_4, \epsilon)
 \end{aligned}$$

Then the string given above was consumed but  $q_4$  was not a *final state*. Then this string does not belong to  $L_2$ .

## Answer 2

$$L_3 = 1^*0(1 \cup 0)^*0(1 \cup 0)^*1$$

## Answer 3

a.

*Countable.*

Take the finite alphabet  $\Sigma$  and say  $Q$  is the set of states of our automaton. We know for any finite alphabet, *Kleene Star* is countably infinite by definition then  $\Sigma^*$  is also countably infinite. Also we know that  $\Sigma^*$  is the set of all strings over  $\Sigma$ .

Since we are talking about a "*finite*" automaton then  $Q$  is finite.

*Configurations* within an automata over these strings are of the form  $(q_i, w_k)$  where  $q_i \in Q$  and  $w_k \in \Sigma^*$ , so  $(q_i, w_k) \in Q \times \Sigma^*$ . Since  $Q$  is finite and  $\Sigma^*$  is countable then their cartesian product is also countable. Hence set of configurations of a finite automaton on all strings over  $\Sigma$  is countable because it is a subset of  $Q \times \Sigma^*$  (*Note: This is because it is not a necessary that every string matches with every state.*). As a result, it is the same as saying that *"the set of configurations of a finite automaton on all strings over finite alphabet is countable."*

**b.**

**Note:** If number of symbols in universe is assumed to uncountable, then the answer will be uncountable since in this case strings over this uncountable set is also uncountable and so all finite automata on these strings are. Hence configurations of all these finite automata are also uncountable since in this case configurations will be from a cartesian product, one of which contributing this cartesian product is an uncountable set.

*Countable.*

We know number of regular expressions on a language over a finite alphabet is countably infinite (**see (\*)1 below**). Also we know for any automaton, we have only one corresponding regular expression over this language. Since regular expression are countable and there exists a 1 – 1 correspondence between regular expressions and finite automata over a certain language, then we can count finite automata. We have already proven that we could count the number of configurations of **"a"** finite automata on all strings over all possible finite alphabets (**see (\*)2 below**). Our finite automata are countably infinite. Hence we can say that *"the set of configurations of all finite automata on all strings over all possible finite alphabets is countable."*

**(\*1)** Regular expressions over a finite alphabet  $\Sigma$  is just *Kleene Star* of a finite set of symbols such that

$$\Sigma \cup \{\cup, *, (, )\}$$

**(\*2) 1) Assuming that the number of symbols in universe, which can be used to create an alphabet, is finite**, number of possible finite alphabets are countable since alphabets are subsets of symbols. Say we have  $n$  many different alphabets. Assume a new alphabet  $\Sigma_{new}$  and say  $\Sigma_{new} = \cup_{i=1}^n \Sigma_i$  where every  $\Sigma_i$  is representing a possible alphabet in the universe. So  $\Sigma_{new}$  is finite since it is the construction of finite number of unions of finite alphabets. Also note that  $\Sigma_{new}^* = \cup_{i=1}^n \Sigma_i^*$ . This is because if we sort the alphabets  $\Sigma_i$  from smaller to larger cardinality, then we can say  $\Sigma_n = \Sigma$ . Also, the strings over other smaller alphabets  $\Sigma_i^*$  are already included in  $\Sigma_n^*$  since  $\Sigma_i^* = (\cup_{k=1}^i \alpha_k)^* \subseteq (\cup_{k=1}^n \alpha_k)^* = \Sigma_n^*$  where each  $\alpha$  represents a symbol.

We know for any finite alphabet,  $\Sigma^*$  is countably infinite by definition then  $\Sigma_{new}^*$  is also countably infinite. Also we know that  $\Sigma^*$  is the set of all strings over  $\Sigma$ .

2) Since we are talking about a *"finite"*, automaton then  $Q$  is finite.

*Configurations* within an automata over these strings are of the form  $(q_i, w_k)$  where  $q_i \in Q$  and  $w_k \in \Sigma_{new}^*$ , so  $(q_i, w_k) \in Q \times \Sigma_{new}^*$ . Since  $Q$  is finite and  $\Sigma_{new}^*$  is countable then their cartesian product is also countable. Hence set of configurations of a finite automaton on all strings over  $\Sigma_{new}$  is countable because it is a subset of  $Q \times \Sigma_{new}^*$  (*Note: This is because it is not a necessary that every string matches with every state.*). As a result, it is the same as saying that "the set of configurations of a finite automaton on all strings over all possible finite alphabets is countable."

## Answer 4

a.

$$E(q_0) = q_0$$

$$E(q_1) = q_1, q_4$$

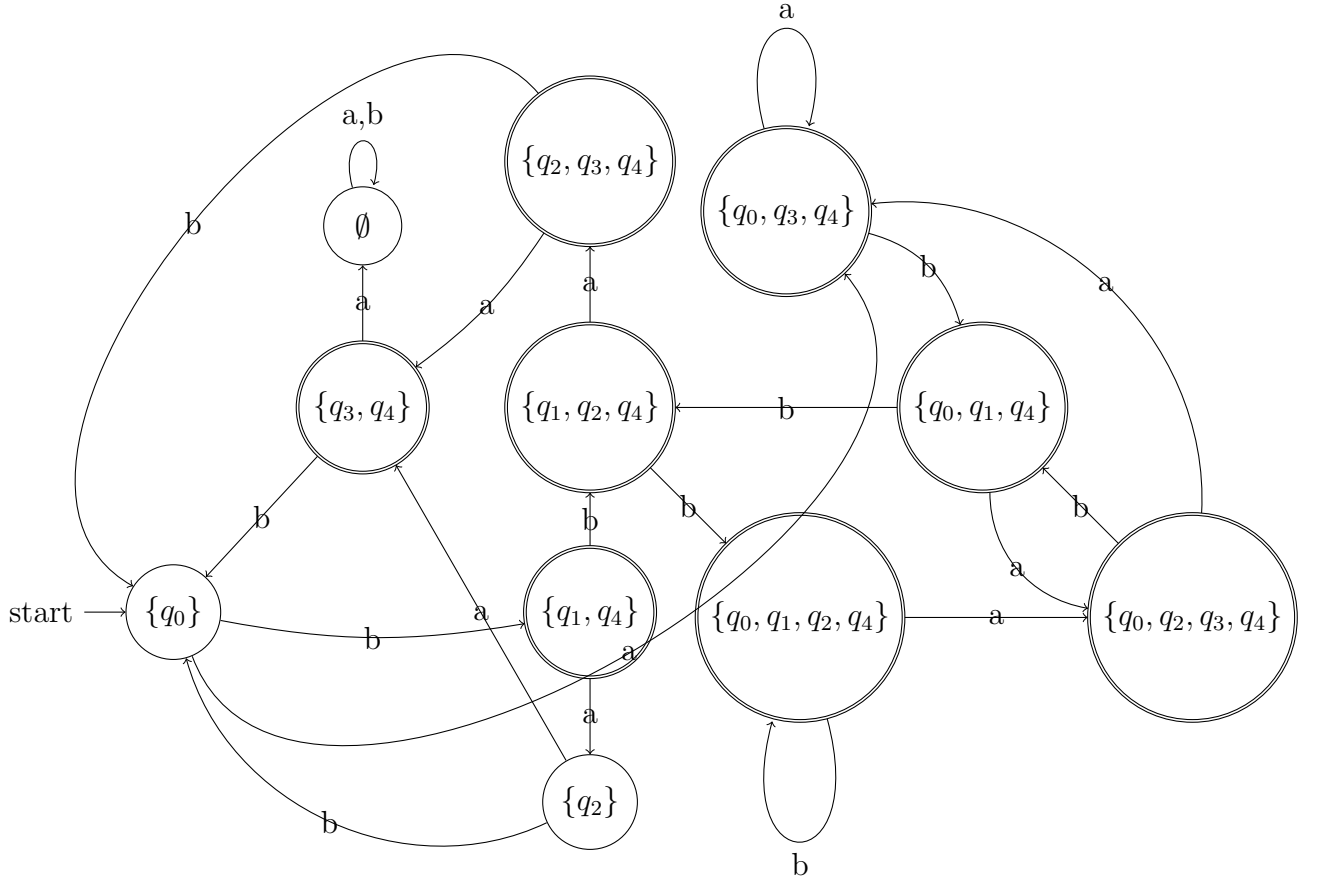
$$E(q_2) = q_2$$

$$E(q_3) = q_3, q_4$$

$$E(q_4) = q_4$$

Where  $E(q_0) = q_0$  will be our starting node. Also we can say that if we have  $E(q_4) = q_4$  in a subset in DFA, then this subset is one of the accepting states.

$$\begin{aligned}
\delta'(\{q_0\}, a) &= E(q_0) \cup E(q_3) &= \{q_0, q_3, q_4\} \\
\delta'(\{q_0\}, b) &= E(q_1) &= \{q_1, q_4\} \\
\delta'(\{q_0, q_3, q_4\}, a) &= E(q_0) \cup E(q_3) &= \{q_0, q_3, q_4\} \\
\delta'(\{q_0, q_3, q_4\}, b) &= E(q_0) \cup E(q_1) &= \{q_0, q_1, q_4\} \\
\delta'(\{q_1, q_4\}, a) &= E(q_2) &= \{q_2\} \\
\delta'(\{q_1, q_4\}, b) &= E(q_1) \cup E(q_2) &= \{q_1, q_2, q_4\} \\
\delta'(\{q_0, q_1, q_4\}, a) &= E(q_0) \cup E(q_2) \cup E(q_3) &= \{q_0, q_2, q_3, q_4\} \\
\delta'(\{q_0, q_1, q_4\}, b) &= E(q_1) \cup E(q_2) &= \{q_1, q_2, q_4\} \\
\delta'(\{q_2\}, a) &= E(q_3) &= \{q_3, q_4\} \\
\delta'(\{q_2\}, b) &= E(q_0) &= \{q_0\} \\
\delta'(\{q_1, q_2, q_4\}, a) &= E(q_2) \cup E(q_3) &= \{q_2, q_3, q_4\} \\
\delta'(\{q_1, q_2, q_4\}, b) &= E(q_0) \cup E(q_1) \cup E(q_2) &= \{q_0, q_1, q_2, q_4\} \\
\delta'(\{q_0, q_2, q_3, q_4\}, a) &= E(q_0) \cup E(q_3) &= \{q_0, q_3, q_4\} \\
\delta'(\{q_0, q_2, q_3, q_4\}, b) &= E(q_0) \cup E(q_1) &= \{q_0, q_1, q_4\} \\
\delta'(\{q_3, q_4\}, a) &= \{\} &= \{\} \\
\delta'(\{q_3, q_4\}, b) &= E(q_0) &= \{q_0\} \\
\delta'(\{q_2, q_3, q_4\}, a) &= E(q_3) &= \{q_3, q_4\} \\
\delta'(\{q_2, q_3, q_4\}, b) &= E(q_0) &= \{q_0\} \\
\delta'(\{q_0, q_1, q_2, q_4\}, a) &= E(q_0) \cup E(q_2) \cup E(q_3) &= \{q_0, q_2, q_3, q_4\} \\
\delta'(\{q_0, q_1, q_2, q_4\}, b) &= E(q_0) \cup E(q_1) &= \{q_0, q_1, q_4\}
\end{aligned}$$



**b.**

Since *empty* string cannot be accepted by our language, then we should include it by concatenating  $L$  to  $L^*$  such that  $L^+ = LL^*$ . Also, we can find  $L$  by sequentially removals of paths. At the end, we have only two states, some loops and some paths which are connecting starting state with accepting state.

$$([bb^*(a \cup b)b \cup bb^*(a \cup b)ab \cup a(b \cup \emptyset)]^*(bb^*(\emptyset \cup (a \cup b)a) \cup a)([bb^*(a \cup b)b \cup bb^*(a \cup b)ab \cup a(b \cup \emptyset)]^*(bb^*(\emptyset \cup (a \cup b)a) \cup a))^*$$

**c.**

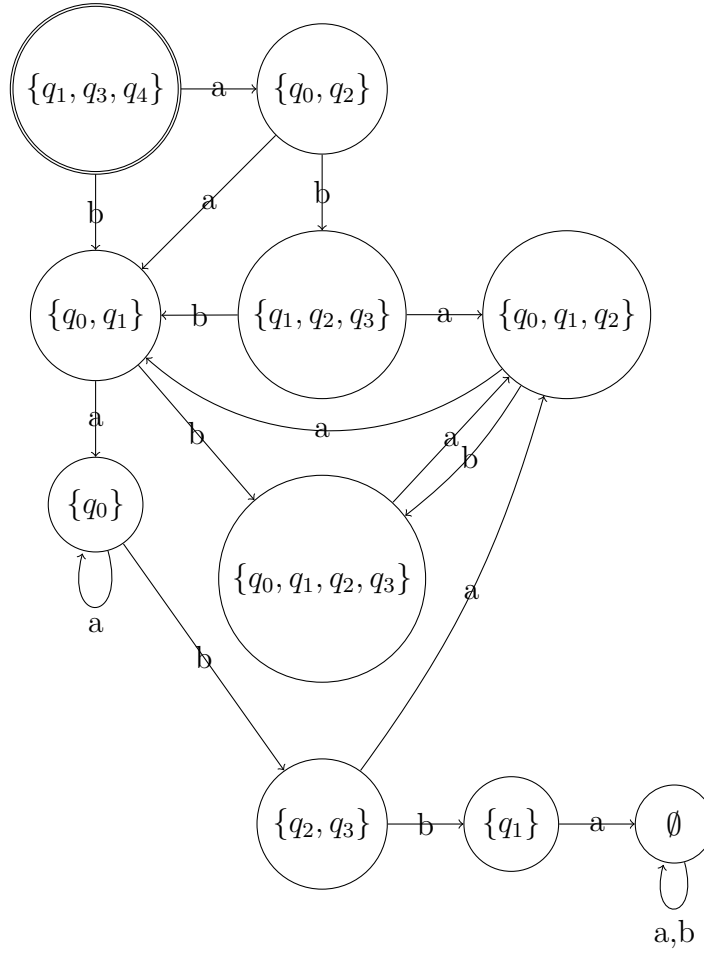
We should first reverse edges. Also we should make  $q_0$  accepting node and  $q_4$  initial node. Then we can change the obtained NFA to DFA by subset construction algorithm.

Now apply subset construction algorithm.

$$\begin{aligned}
E(q_0) &= \{q_0\} \\
E(q_1) &= \{q_1\} \\
E(q_2) &= \{q_2\} \\
E(q_3) &= \{q_3\} \\
E(q_4) &= \{q_1, q_3, q_4\}
\end{aligned}$$

Where  $E(q_4) = \{q_1, q_3, q_4\}$  will be our starting node. Also we can say that if we have  $E(q_0) = q_0$  in a subset in DFA, then this subset is one of the accepting states.

$$\begin{aligned}
\delta'(\{q_1, q_3, q_4\}, a) &= E(q_0) \cup E(q_2) &= \{q_0, q_2\} \\
\delta'(\{q_1, q_3, q_4\}, b) &= E(q_1) \cup E(q_0) &= \{q_0, q_1\} \\
\delta'(\{q_0, q_2\}, a) &= E(q_0) \cup E(q_1) &= \{q_0, q_1\} \\
\delta'(\{q_0, q_2\}, b) &= E(q_1) \cup E(q_2) \cup E(q_3) &= \{q_1, q_2, q_3\} \\
\delta'(\{q_0, q_1\}, a) &= E(q_0) &= \{q_0\} \\
\delta'(\{q_0, q_1\}, b) &= E(q_0) \cup E(q_1) \cup E(q_2) \cup E(q_3) &= \{q_0, q_1, q_2, q_3\} \\
\delta'(\{q_1, q_2, q_3\}, a) &= E(q_0) \cup E(q_1) \cup E(q_2) &= \{q_0, q_1, q_2\} \\
\delta'(\{q_1, q_2, q_3\}, b) &= E(q_0) \cup E(q_1) &= \{q_0, q_1\} \\
\delta'(\{q_0\}, a) &= E(q_0) &= \{q_0\} \\
\delta'(\{q_0\}, b) &= E(q_2) \cup E(q_3) &= \{q_2, q_3\} \\
\delta'(\{q_0, q_1, q_2, q_3\}, a) &= E(q_0) \cup E(q_1) \cup E(q_2) &= \{q_0, q_1, q_2\} \\
\delta'(\{q_0, q_1, q_2, q_3\}, b) &= E(q_0) \cup E(q_1) \cup E(q_2) \cup E(q_3) &= \{q_0, q_1, q_2, q_3\} \\
\delta'(\{q_0, q_1, q_2\}, a) &= E(q_0) \cup E(q_1) &= \{q_0, q_1\} \\
\delta'(\{q_0, q_1, q_2\}, b) &= E(q_0) \cup E(q_1) \cup E(q_2) \cup E(q_3) &= \{q_0, q_1, q_2, q_3\} \\
\delta'(\{q_2, q_3\}, a) &= E(q_0) \cup E(q_1) \cup E(q_2) &= \{q_0, q_1, q_2\} \\
\delta'(\{q_2, q_3\}, b) &= E(q_1) &= \{q_1\} \\
\delta'(\{q_0, q_1, q_2\}, a) &= E(q_0) \cup E(q_1) &= \{q_0, q_1\} \\
\delta'(\{q_0, q_1, q_2\}, b) &= E(q_0) \cup E(q_1) \cup E(q_2) \cup E(q_3) &= \{q_0, q_1, q_2, q_3\} \\
\delta'(\{q_1\}, a) &= \emptyset &= \emptyset \\
\delta'(\{q_1\}, b) &= E(q_0) \cup E(q_1) &= \{q_0, q_1\}
\end{aligned}$$



## Answer 5

We can refer  $L$  as  $L = (L_1 L_3 \cap \overline{L_2}) \cup L_4$  by set operations.

Since regular languages are closed under concatenation then  $L_1 L_3$  is regular. Since regular languages are closed under complementation then  $\overline{L_2}$  is also regular. Since regular languages are closed under intersection then  $L_1 L_3 \cap \overline{L_2}$  is also regular. Since regular languages are closed under union, then  $L = (L_1 L_3 \cap \overline{L_2}) \cup L_4$  is also regular.

Say  $L_1 = (K_1, \Sigma, \Delta_1, q_{01}, F_1)$ ,  $L_2 = (K_2, \Sigma, \Delta_2, q_{02}, F_2)$ .  $L_3 = (K_3, \Sigma, \Delta_3, q_{03}, F_3)$ .  
 $L_1 L_3 = (K_1 \cup K_3, \Sigma, \Delta_1 \cup \Delta_3 \cup (F_1 \times \{q_{03}\}), q_{01}, F_3)$ .  $\overline{L_2} = (K_2, \Sigma, \Delta_2, q_{02}, K_2 - F_2)$ .

$$\begin{aligned}
 L_1 L_3 \cap \overline{L_2} &= \overline{L_1 L_3} \cup L_2 \\
 &= (K_1 \cup K_3, \Sigma, \Delta_1 \cup \Delta_3 \cup (F_1 \times \{q_{03}\}), q_{01}, (K_1 \cup K_3) - F_3) \cup (K_2, \Sigma, \Delta_2, q_{02}, F_2) \\
 &= (K_1 \cup K_3 \cup K_2 \cup \{q_{013}\}, \Sigma, \Delta_1 \cup \Delta_3 \cup (F_1 \times \{q_{03}\}) \\
 &\quad \cup \Delta_2 \cup \{(q_{013}, e, q_{01}), (q_{013}, e, q_{02})\}, q_{013}, ((K_1 \cup K_3) - F_3) \cup F_2)
 \end{aligned}$$



## Answer 6

Take an alphabet  $\Sigma$ . Define the languages  $L_1, L_2$  and  $L_3$  over the alphabet  $\Sigma$  and specifically  $L_2$  is not to be a regular language. Assume  $L_1 = \Sigma^*$ . Whatever  $L_3$  and  $L_2$  is,  $L_1 \subseteq \Sigma^*$  hence  $L_1 \cup L_2 \cup L_3 = \Sigma^*$ , which is a regular language. However  $L_2 L_3$  is not a regular language. So proof is done, we found a contradiction. It is not necessary that  $L_2 L_3$  is a regular language.

## Answer 7

a.

Name set of all strings that constitute valid configurations as  $L$ .

$$L = \{w : w = \underbrace{11 \dots 11}_{m \text{ many } 1s} 0 \underbrace{11 \dots 11}_{n \text{ many } 1s} 0 \underbrace{11 \dots 11}_{k \text{ many } 1's}, \forall (m, n \in N) \ k = m + n\}$$

b.

*Pumping Lemma* holds for  $L$  if:

$$\forall x = uvw \in L, |x| \geq l \text{ where } u, v, w \in \{0, 1\}^*, |uv| \leq l, |v| \geq 1 \text{ and } \forall i \geq 0, uv^i w \in L.$$

Choose  $|uv| \leq l$  and  $|v| \geq 0$ , which means  $uv = 1^p$  for some  $0 < p \leq l$  so that  $y = 1^r$  for some  $1 < r \leq p$  and  $w = 01^t 01^{p+t}$ . Now pump 1s so that  $uv^2 w = 1^{p-r+2r} 01^t 01^{p+t} = 1^{p+r} 01^t 01^{p+t}$  and which cannot be accepted by our language. This is a contradiction,  $L$  is not regular.