Student Information

Full Name : Yaşar Cahit Yıldırım
Id Number : 2310647

Answer 1

First let's examine the given sets.
A_k is finite since there are $2^{k+1} - 1$ words in it.
B is countably infinite, because we can enumerate words by their lengths.
C is uncountably infinite by Cantor's diagonalization theorem.

a.

a_{280} is a finite set and B is a countably infinite set, so their union is countably infinite. Since C is uncountably infinite and the difference of an uncountably infinite set with a countably infinite set is uncountably infinite too. So this set is **uncountably infinite**.

b.

C is the powerset of all the words that can be written with alphabet $\{0, 1\}$ so intersection of any set on this alphabet with C is the set itself. A_7 contains all the words with length $|w| \geq 7$ and its cardinality is $2^8 - 1$. B^* is in any length so in order to take its intersection with A it is sufficient to just look words that are shorter than 8. This starts with empty string and counting its possibilities with multiple 0's at the start and various ends after the 01 part gives us $2^7 - 7$ words. Since this set is smaller that A_7 we can take this as our result. The answer is **finite** with cardinality $\mathbf{2^7 - 7}$.

c.

Since C is the powerset of $\{0, 1\}^*$ it includes all of its subsets. And union of all these subsets gives us $\{0, 1\}^*$ again. $A_2 \times B$ is a set of tuples, so there is no intersection of it with $\bigcup C$. The difference operator does not change anything and the answer is $|\{0, 1\}^*|$ which is **countably infinite**, by mapping words by their lengths to the integers.

d.

$\bigcup C = \{0, 1\}^* = \bigcup_{k=1}^{\infty} A_k$ so cardinality of $(\bigcup C \cup \bigcup_{k=1}^{\infty} A_k) \setminus \{0, 1\}^*$ is $\mathbf{0}$. It is an empty set.

Answer 2

For parts (**a**) and (**b**), we can use The Fundamental Principle of Counting.

a.

(number of transition functions) * (number of initial states) * (number of final state sets)

$$(4^2)^4 * 4 * (2^4) = \mathbf{2^{22}}$$

b.

(number of transition relations) * (number of initial states) * (number of final state sets)

$$(2^{3*4})^4 * 4 * (2^4) = \mathbf{2^{54}}$$

c.

The difference is solely because of the formal definitions of a DFA and NFA. While **DFA** requires a **transition function** for each configuration, **NFA** requires a **transition relation** and it also transitions empty symbol moves. So at each state of the automaton, there are much more possibilities to account in a NFA.

d.

Every FA corresponds to a regular language, and by using minimization we can produce the same language with an automaton that has fewer states. We also know that every NFA can be converted to DFA through the process of subset construction algorithm. So we can say that the number of different languages M recognizes L_x is less than or equal to the minimum of the number of DFA and the number of NFA, which is always can be taken as number of DFA.

$$\boldsymbol{L_x \leq min(2^{54}, 2^{22})}$$

Answer 3

a.

$$L_1 = \mathcal{L}(\alpha_1) \text{ where } \alpha_1 \text{ is } \big((0^*)10(0^*)10(0^*)10(0^*)10(0^*)\big)^*.$$

b.

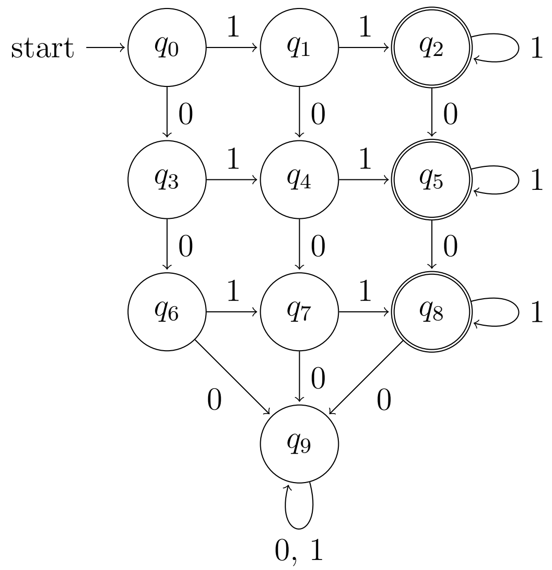
$$L_2 = \mathcal{L}(\alpha_2) \text{ where } \alpha_2 \text{ is } \Big(0\Big(00 \cup \big(01(01)^*00\big)\Big)^*\Big) \cup \big(1(01)^*\big).$$

c.

$$L_3 = \mathcal{L}(\alpha_3) \text{ where } \alpha_3 \text{ is } \Big(\big(1 \cup 01 \cup 00(1 \cup 01)^*00\big)^*\Big) \cup (e \cup 0) \text{ and } e \text{ is empty string.}$$

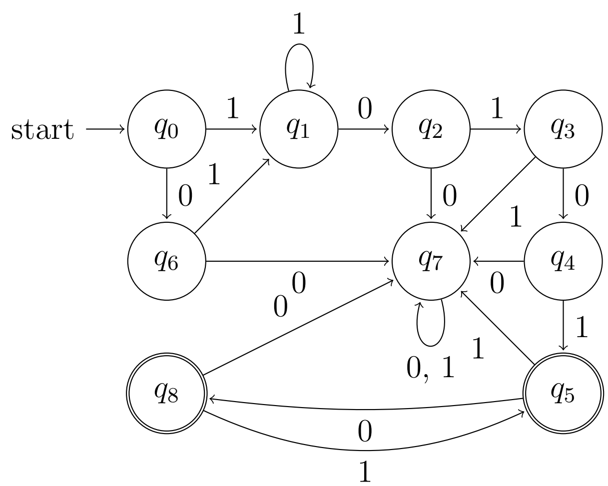
Answer 4

a.



$$K = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8, q_9\}$$
$$\Sigma = \{0, 1\}$$
$$s = q_0$$
$$F = \{q_2, q_5, q_8\}$$
$$where\ M = (K, \Sigma, \delta, s, F)$$

b.



$$K = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8\}$$
$$\Sigma = \{0, 1\}$$
$$s = q_0$$
$$F = \{q_5, q_8\}$$
$$where\ M = (K, \Sigma, \delta, s, F)$$

Answer 5

a.

$$(q_0, abaa) \vdash_N (q_2, abaa) \vdash_N (q_2, baa) \vdash_N (q_2, aa) \vdash_N (q_4, a)$$
$$\vdash_N (q_1, e)$$

$(q_0, abaa) \vdash_N^* (q_1, e)$ and q_1 is a final state, so $w_1 = abaa$ is a word N accepts.

b.

$$(q_0, babb) \vdash_N (q_1, abb) \ \vdash_N (q_1, bb) \tag{1}$$
$$(q_0, babb) \vdash_N (q_2, babb) \vdash_N (q_2, abb) \vdash_N (q_2, bb) \vdash_N (q_2, b) \vdash_N (q_2, e) \tag{2}$$
$$(q_0, babb) \vdash_N (q_2, babb) \vdash_N (q_2, abb) \vdash_N (q_4, bb) \vdash_N (q_4, b) \vdash_N (q_4, e) \tag{3}$$
$$(q_0, babb) \vdash_N (q_3, abb) \ \vdash_N (q_1, bb) \tag{4}$$
$$(q_0, babb) \vdash_N (q_3, abb) \ \vdash_N (q_4, abb) \vdash_N (q_1, bb) \tag{5}$$
$$(q_0, babb) \vdash_N (q_3, abb) \ \vdash_N (q_4, abb) \vdash_N (q_2, bb) \vdash_N (q_2, b) \vdash_N (q_2, e) \tag{6}$$

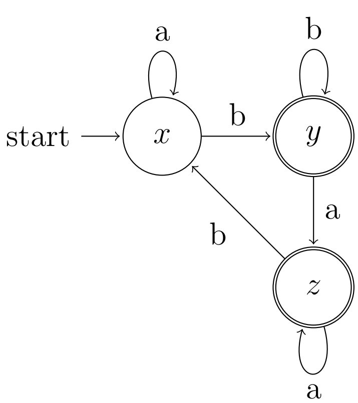
On first, fourth and fifth reads, there are no further configurations. On second, third and sixth reads, the word is completely read but the finishing states are not final states.

We can conclude that $w_2 = babb$ is not a word N accepts.

Answer 6

Starting state is q_0 so we are starting the construction from q_0. And F' will be the set of new states that include NFA's accepting states, which are q_3 and q_4. Let x, y, z be states of the new DFA.

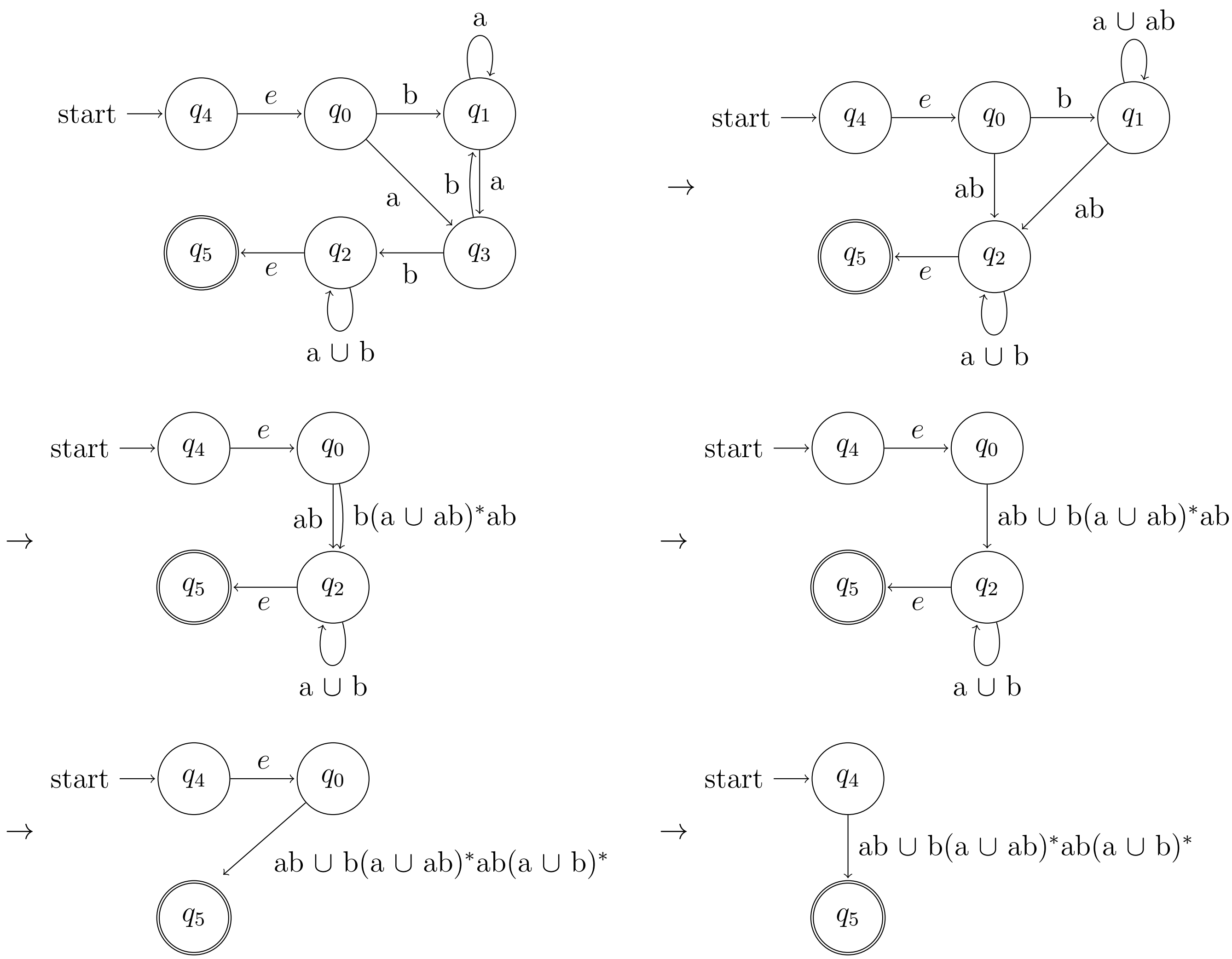
$$\begin{aligned}
E(q_0) &= \{q_0, q_2\} = x \\
\delta'(x, a) &= E(q_0) = x \\
\delta'(x, b) &= E(q_0) \cup E(q_1) \cup E(q_3) = \{q_0, q_1, q_2, q_3\} = y \\
\delta'(y, a) &= E(q_0) \cup E(q_4) = \{q_0, q_2, q_3, q_4\} = z \\
\delta'(y, b) &= E(q_0) \cup E(q_1) \cup E(q_3) = \{q_0, q_1, q_2, q_3\} =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



$$\begin{aligned}
K' &= \{x, y, z\} \\
\Sigma &= \{a, b\} \\
s' &= x \\
F' &= \{y, z\} \\
where \ \ M &= (K', \Sigma, \delta', s', F')
\end{aligned}$$

Answer 7

First let's draw the initial GFA where e is empty string, and eliminate states one by one.

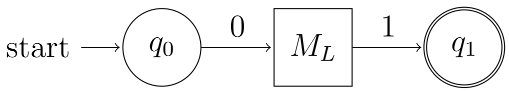


So, $\mathbf{R} = \text{ab} \cup \text{b}(\text{a} \cup \text{ab})^*\text{ab}(\text{a} \cup \text{b})^*$

Answer 8

If 0 and 1 are elements of an alphabet Σ, then we can say that they are regular expressions since we know that each member of the alphabet Σ is a regular expression. They can be shown as $\mathcal{L}(0)$ and $\mathcal{L}(1)$. If L is a regular language then we can show it as $\mathcal{L}(L)$, too. Using the definition of regular expression we can conclude $H = \mathcal{L}(0)\mathcal{L}(L)\mathcal{L}(1)$. H is a regular language.

We can define a new transition relation δ' for the new FA such that the initial state q_0 yields M_L's initial state with input 0, and M_L's all final states yields q_1 with input 1. Assume K is the set of states for FA M_L



$$M_H = (\{q_0, q_1\} \cup K, \Sigma, \delta', q_0, q_1)$$

With state elimination we find $M_H = 0M_L1$; so $L(0M_L1) = 0w1$ where $w \in \Sigma^*$ and is accepted by M_L and $M_H = H$.

Answer 9

a.

Assume L is regular and let p be its pumping length. $w \in L$ where $w = 1^t0^p1^{2^{p+p!}}$ and $|w| \geq p$. Using pumping lemma we can divide w as xyz where $x = 1^t0^a, y = 0^b, z = 0^c1^{2^{p+p!}}$ and as pummping lemma specifies $b \geq 1$ with $a + b + c = p$.

Now, examine the string $w' = xy^{i+1}z$ where $i = \frac{p!}{b}$. (Note that $p!$ is divisible by b since it contains b as a factor.) Then $y^i = 0^{p!}$, $y^{i+1} = 0^{p!+b}$. Now we have,

$$w' = 1^t0^{p!+a+b+c}1^{2^{p+p!}} = 1^t0^{p!+p}1^{2^{p+p!}}.$$

We know that $w' \notin L$ since $p + p! = p! + p$ and thus this is a contradiction. We conclude that L is not a regular language by pumping lemma.