Student Information

Full Name : Doruk Gerçel
Id Number : 2310027

Answer 1

a.

From the lemma we know that the number of $\sum^*$ is equal to a countably infinite number N. Also we know that number of subsets of this number is equal to 2^N which is uncountably infinite set. So we say that C is an uncountably infinite set.(By defintion C is the set of all subset of words that can be created over an alphabet.) In part a, the expression means the set of all subset of words that can be created over an alphabet minus the words that are shorter than length 280 and the every expression that can be created according to the statement given by B, and their union doesn't contain all the subset of words that can be created over the alphabet. Therefore this expression is uncountably infinite.

b.

This expression states that the intersection of word that are not longer than length 7, the different number of concatenation of the expression stated in B and the set of all subsets of the words that can be created over the alphabet. Therefore the set can be actually formed according to this constraints easily, therefore this expression is countably finite.

c.

d.

The expression $\cup C$ actually expresses every word that can be formed by the alphabet, and we take the union with the expression A_K which is a subset of the first part. As we take their union we can just take the bigger set and say that the first part only consists of $\cup C$. $\{0, 1\}^*$ is the every word that can be created by using the alphabet. As the first part and second part are actually equal, when we eliminate the second part from the first part we get an empty set.

Answer 2

a.

According to definition of DFA, we must have a single initial state, and we the transitions are controlled by a transition function. Therefore transitions must show a function property.(0,1 can move from one state to only one state.) Also there is no specification about the final states set. So, the number of,

Initial state: 4
Final state: C(4,0)+C(4,1)+C(4,2)+C(4,3)+C(4,4) = 16
Transitions:(4.4).(4.4).(4.4).(4.4) = $(4.4)^4$
So when we multiply them the number of DFA is 2^{22}.

b.

According to defintion of NFA, we must have a single initial state, but the transitions doesn't show function property. A single character can direct to multiple states, and also empty string can direct to a state aswell. Also there is no specification about the final states set. So number of,
Initial state: 4
Final state: C(4,0)+C(4,1)+C(4,2)+C(4,3)+C(4,4) = 16
Transisitons: For a single state there can be 16 connections, and we check if there is a connection or not, and we obtain 16.16.16 and we check that for 4 states and we obtain (2^{16})).
When we multiply them we obtain 2^{54}.

c.

The number of DFA and NFA occured because, DFA transisitons are controlled by a transition funtion. Every element goes to a single state and empty string doesn't go to any state. But in NFA there is no transition function so we calculated it's number by checkin either is there any connection between states or not.

d.

DFA recognizes less language than NFA as there can be less number of automatas that are created by DFA. This actually shows that transition function of DFA is nothing but a subset of the transition set of NFA.

Answer 3

a.

The sum of the interior elements that the regular expression includes must be the multiple of 2 and 4, therefore it must include 1's as multiples of 4 in it. Also the other condition states that non of the two consecutive elements sum can be 2, therefore there can't be any 2 consecutive 1's in our regular expression. Also, as we won't reach any element of an empty string we disclude empty string condition.Therefore,
$L_1 = ((00^*100^*100^*100^*1)^* \cup (100^*100^*100^*100^*)^*) \setminus e$

b.

The length of our regular expression must be odd. Also the other condition states that every even indexed element of the string must be 0. So odd indexed elements can be 0 or 1. Therefore,

$L_2 = ((1 \cup 0).0)^*.(1 \cup 0)$

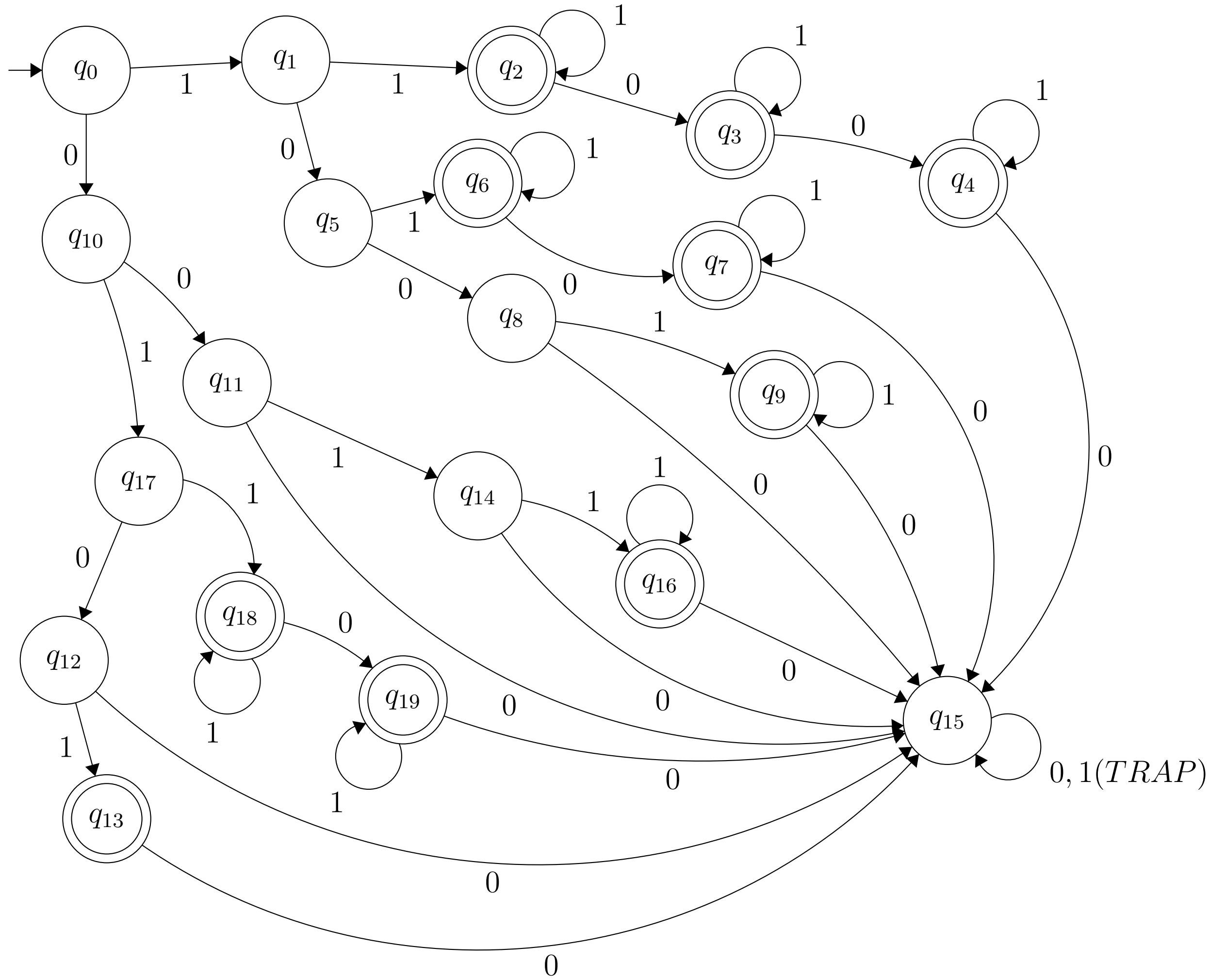
c.

The condition states there must be even numbered occurrence of 00 substring in the string. Therefore to perform this condition, there must be (4*i) number of 0's or (4*i+1) number of 0's in our string. Therefore,
$L_3 = (1^*.00.1^*.00.1^*)^*.(0 \cup 1^*)$

Answer 4

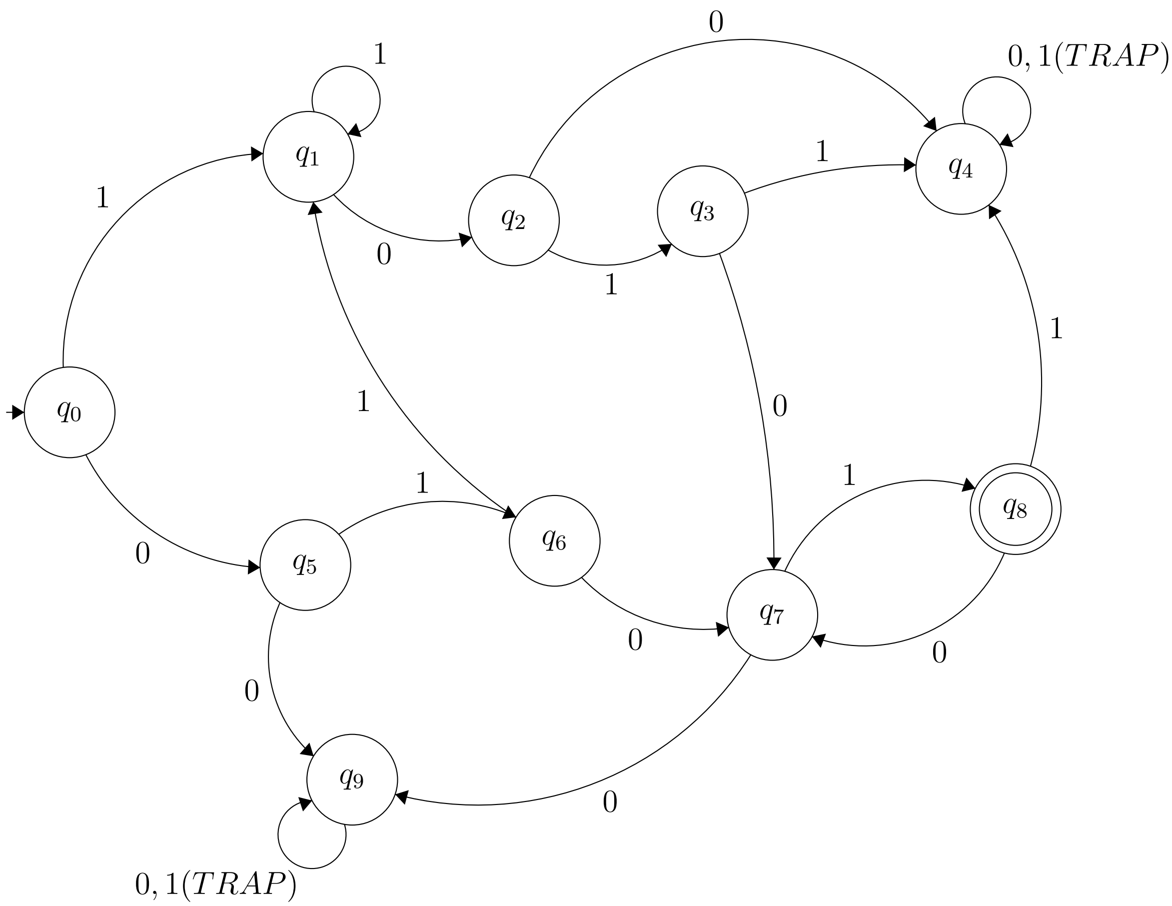
a.

$M(L_1) = (K, \{0, 1\}, \delta, s, F)$
$K = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8, q_9, q_{10}, q_{11}, q_{12}, q_{13}, q_{14}, q_{15}, q_{16}, q_{17}, q_{18}, q_{19}$
$s = q_0$
$F = \{q_2, q_3, q_4, q_6, q_7, q_9, q_{13}, q_{16}, q_{18}, q_{19}\}$



b.

$M(L_2) = (K, \{0, 1\}, \delta, s, F)$
$K = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8, q_9\}$
$s = q_0$
$F = \{q_8\}$



Answer 5

a.

To perform the reflexive transitive closure, when we start with the initial state in the automata, when we follow the transitions, we should end in a final state. Therefore we can state that our language is in L(N).

$$\begin{aligned}(q_0, abaa) &\vdash_M (g_2, abaa)\ [(q_0, e, q_2)] \\ &\vdash_M (g_2, baa)\ [(q_2, a, q_2)] \\ &\vdash_M (g_2, aa)\ [(q_2, b, q_2)] \\ &\vdash_M (g_4, a)\ [(q_2, a, q_4)] \\ &\vdash_M (g_1, e)\ [(q_4, a, q_1)]\end{aligned}$$

As we see abaa ends in the final state q_1, by following the transitions, therefore it is L(N).

b.

The only way for babb expression to be L(N) is that, it ends in one of the final states q_1 or q_3. When we check the transitions of this expression, we can see that the only transitions of b are $(q_0, b, q_1), (q_0, b, q_3), (q_2, b, q_2), (q_4, b, q_4)$. (We checked the element b, because the last letter of the expression is b and there is no way to reach a final state by an empty string.) We will only consider the $(q_0, b, q_1), (q_0, b, q_3)$ transitions as they end up in the final states. (Also we see that the only way to reach from another state to our final states is to read letter b.) Therefore, we can see that, our state before our final transition must be q_0, but there is no possible transition to reach q_0 from another state. So our expression will never end up in a final state, so we conclude that babb is not L(N).

Answer 6

We will define our initial state as $s\prime = E(q_0) = \{q_0, q_2\}$

$(q_0, a, q_0), (q_0, b, q_0), (q_0, b, q_1)$

We didn't consider empty string transition as there is no such type of transition in DFA.

From now on, we are going to form our transition function according to the sets that we obtain, with the states that they reach with the given elements (E(q)'s that they can reach without reading any input). And we will repeat this procedure with the every new set that we obtain, until we can't creat a new set. Procedure for this transition function is as follows.

$\delta\prime(\{q_0, q_2\}, a) = E(q_0) = \{q_0, q_2\}$
$\delta\prime(\{q_0, q_2\}, b) = E(q_0) \cup E(q_1) \cup E(q_3) = \{q_0, q_1, q_2, q_3\}$

$\delta\prime(\{q_0, q_1, q_2, q_3\}, a) = E(q_0) \cup E(q_4) = \{q_0, q_2, q_3, q_4\}$
$\delta\prime(\{q_0, q_1, q_2, q_3\}, b) = E(q_0) \cup E(q_1) \cup E(q_3) = \{q_0, q_1, q_2, q_3\}$

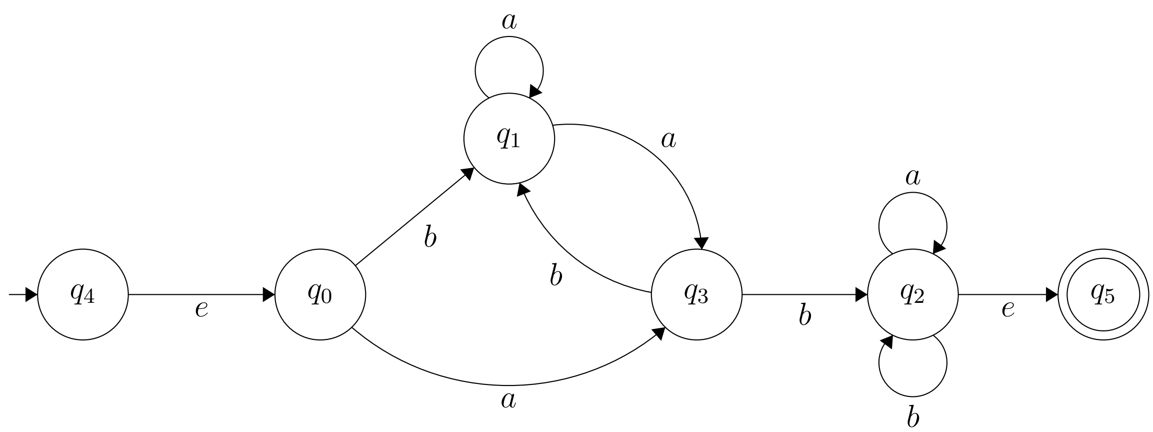
$\delta\prime(\{q_0, q_2, q_3, q_4\}, a) = E(q_0) \cup E(q_4) = \{q_0, q_2, q_3, q_4\}$
$\delta\prime(\{q_0, q_2, q_3, q_4\}, b) = E(q_0) \cup E(q_1) \cup E(q_3) = \{q_0, q_1, q_2, q_3\}$

Our transition function is stated in the above and we determined our initial state at the begining. Our final state in the NFA was $F = \{q_3, q_4\}$. Therefore our final states for DFA is going to be every state that includes q_3 or q_4. So $F\prime = (\{q_0, q_1, q_2, q_3\}, \{q_0, q_2, q_3, q_4\})$. So we know we form our DFA according to the set of states $K\prime$(We formed all of them with the transition function procedure above.) transition function $\delta\prime$, final states $F\prime$, and the initial state $s\prime$ that we determined above. Our $M = (K\prime, \{a, b\}, \delta\prime, s\prime, F\prime)$.

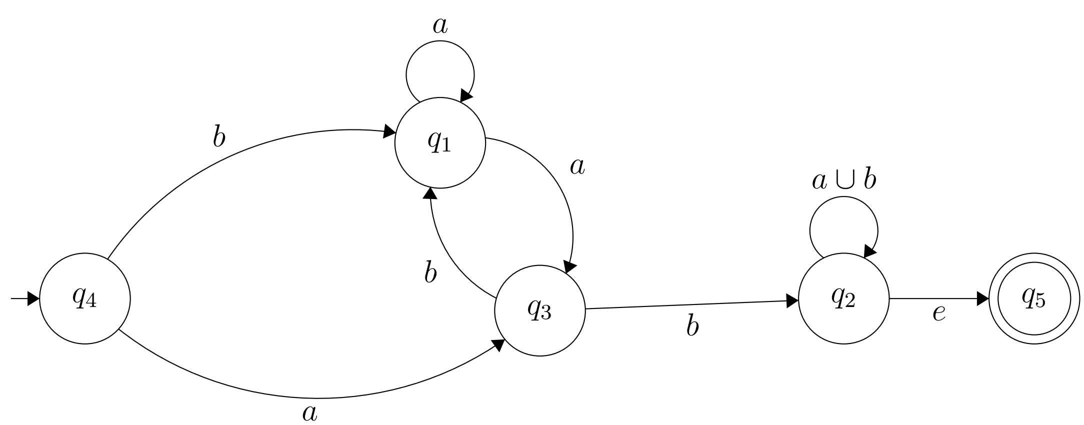
Answer 7

We start our process by adding two new states, which are q_4 and q_5. We have an empty string transition from q_4 to q_0 and q_4 is our new initial state. Also we have an empty string transition from q_2 to q_5 and q_5 is our new final state.

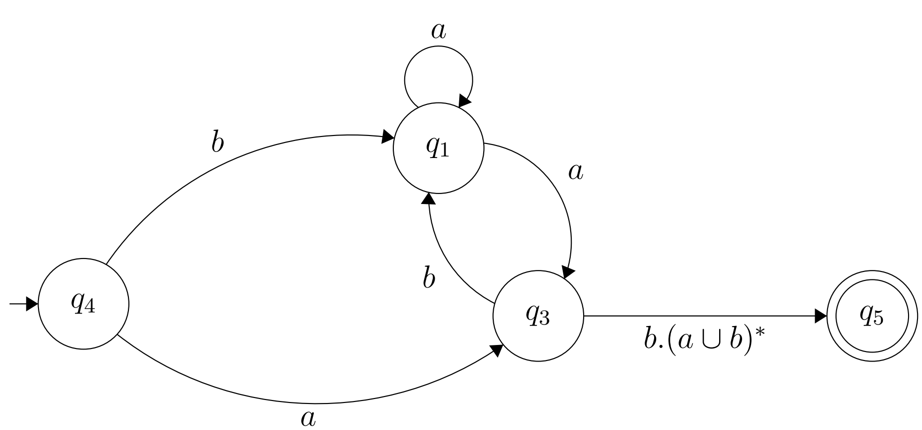
Our automata it follows.



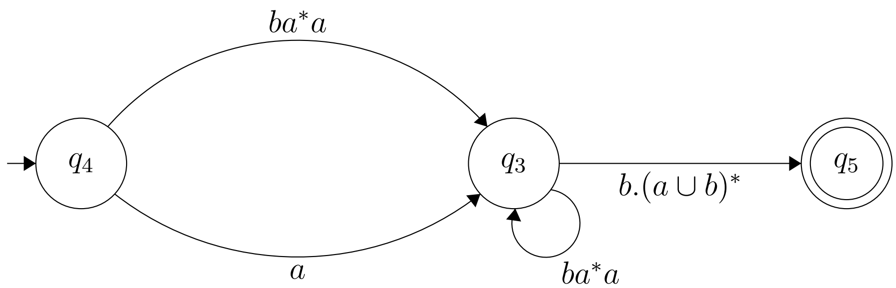
Step 1: We eliminate q_0, and write inner loops of q_2 as $a \cup b$.



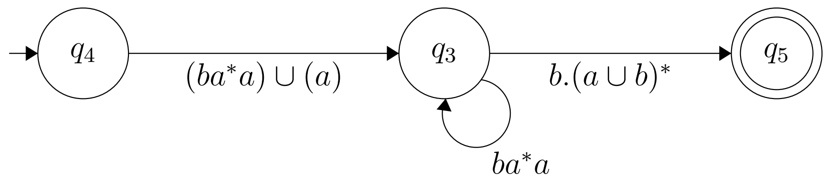
Step 2: We eliminate q_2.



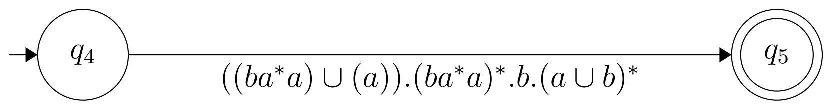
Step 3: We eliminate q_1.



Step 4: We combine the transitions form q_4 to q_3.



Step 5: We eliminate q_3.



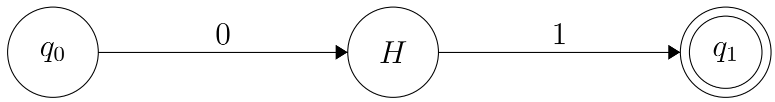
Now we eliminated all the transitions and states except the inital state and the final state. We can write our regular expression as a single transition which is $((ba^*a) \cup (a)).(ba^*a)^*.b.(a \cup b)^*$.

Answer 8

We know that w $\in$ H. Also we will form a language which is A in the form of 0w1. We know that 0w1 is an element of language L therefore we consider the language A as a subset of L. Also let's define another language B in the form of $0 \sum^* 1$. When we take the intersecion of the languages L and B we obtain the language A.$(L \cap B = A)$Therefore when we concenate a state with the transition 0 from the begining of the automata of H and concenate a final statement with the transition 1 we form the automata for the language A. Therefore our automata for A is, $M_A =$ (K = every state of M_H and the concenated initial and final staes, 0,1, $\triangle$, s = concenated state with transition 0, F = concenated state with transition 1).Therefore when L is regular, as it includes the A, A is regular. So as A is regular, H is regular. To show formally,

H = $\{w \mid 0w1 \in L\}$
A = 0w1
B = $0 \sum^* 1$
B $\cap$ L = A

A is,



Answer 9

a.

According to the pumping lemma we divide our word as $w = xy^iz$. Also we must be able to choose our i any number as long as it is bigger than 0.

Take t=3, m=3, n=1 in our term $1^t0^m1^{2^n}$.
Our expression is 11100011.
Choose x = "", y = 11100, z = 011
When we take i = 2, our expression becomes, 1110011100011.($1^x0^y1^z0^a1^b$)
Therefore, this expression doesn't obey the constraints of language L,so L is not a regular language.