

- Please give details of your calculation. A direct answer without explanation is not counted.
- Your answers must be in English.
- Please carefully read problem statements.
- During the exam you are not allowed to borrow others' class notes.
- Try to work on easier questions first.

Problem 1 (15 pts)

1. Consider the following lanagage

$$A = \{0^n 1 \mid n \geq 0\}$$

Give a DFA state diagram for this language (no more than 3 states)

Give the formal definition of your DFA

2. Consider the following lanagage

$$B = \{(10)^n \mid n \geq 0\}$$

Give a DFA state diagram for this language (no more than 3 states)

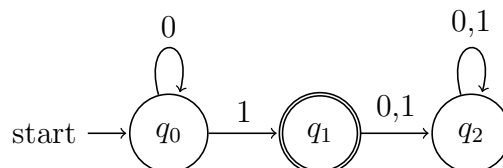
No need to give the formal definition of your DFA

3. Construct the state diagram of $A \cup B$ using the method used in Theorem 1.25 (i.e., the method before we introduced NFA). Please remove unnecessary states in final answer.

No need to give the formal definition of your DFA

Answer

- 1.

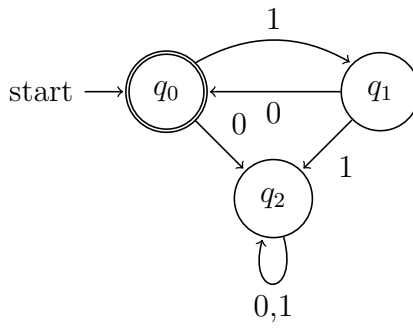


Formal definition: $M = (Q, \Sigma, \delta, q_0, F)$

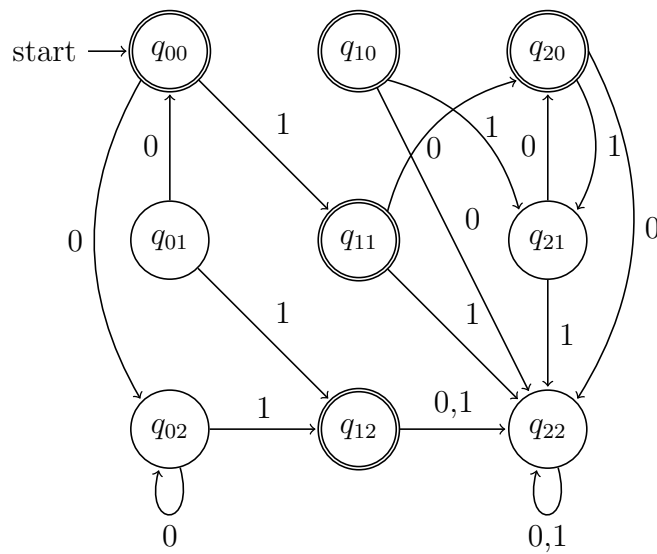
- $Q = \{q_0, q_1, q_2\}$
- $\Sigma = \{0, 1\}$
- $F = \{q_1\}$
- δ is described as

	0	1
q_0	q_0	q_1
q_1	q_2	q_2
q_2	q_2	q_2

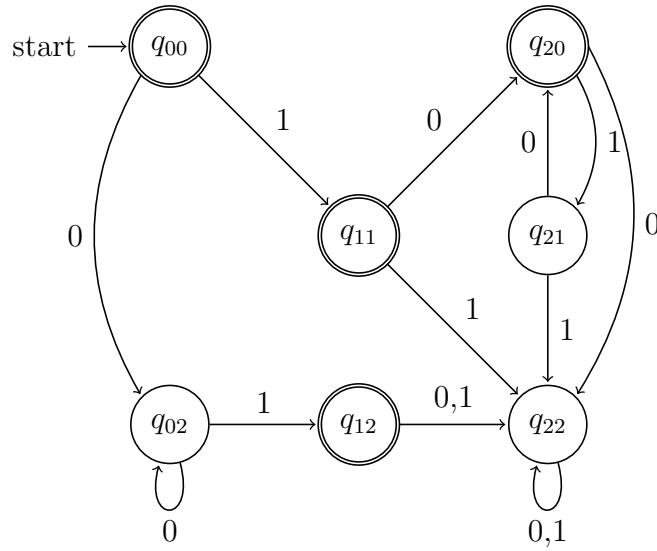
2.



3.



The simplified diagram is



Problem 2 (15 pts)

1. For any regular language A , consider the following way to generate language A^* .
 1. Find a DFA M (Figure 1a) which recognizes language A .
 2. Add an edge from each accepting state to the initial state with empty character ϵ .
 3. Since we need to accept empty string, the initial state should also be an accepting state.
 4. So we get a new NFA (Figure 1b).

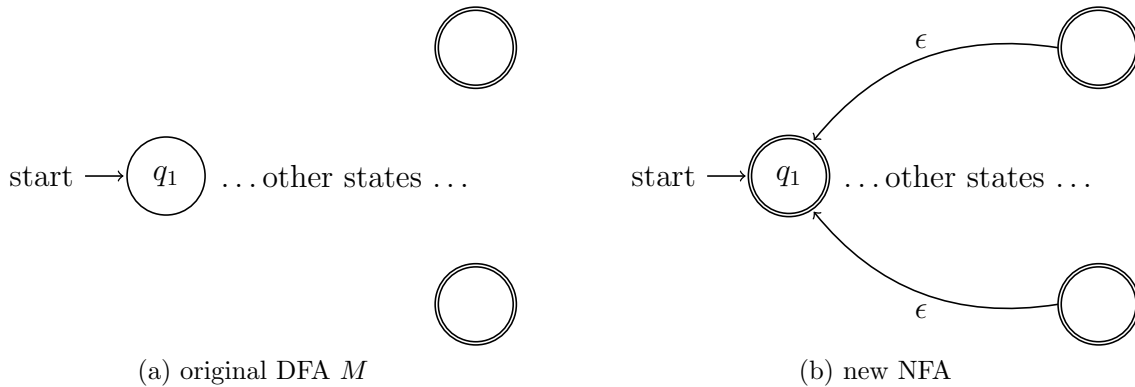


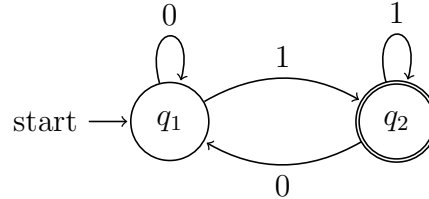
Figure 1: A NFA generated from DFA M by the procedure in Problem 2.

Give a DFA M with number of states = 2, such that the language generated by the procedure above is **not** A^* . You only need to give a state diagram, and you can assume $\Sigma = \{0, 1\}$.

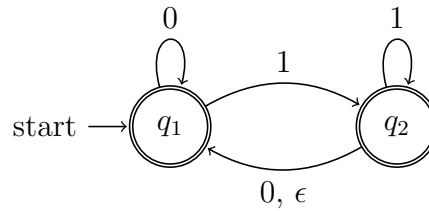
2. For a general DFA $M = (Q_1, \Sigma, \delta_1, q_0, F_1)$, what is the formal definition of the NFA in Figure 1?

Answer

1. The following DFA recognizes Σ^*1 .



Following the procedure, we obtain the following NFA, which accepts any string, but we want it to recognize $(\Sigma^*1)^* = 0 \cup \Sigma^*1$.



2. The new NFA $M' = (Q_1, \Sigma, \delta'_1, q_0, F'_1)$, where

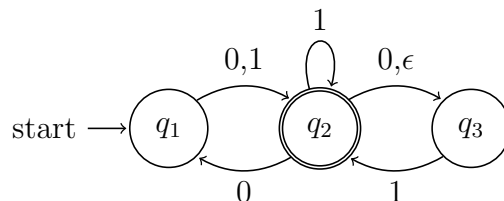
$$F'_1 = F_1 \cup \{q_0\}, \text{ and}$$

$$\delta'_1(q, \sigma) = \begin{cases} \{\delta_1(q, \sigma)\} & \text{if } \sigma \neq \epsilon \\ \{q_0\} & \text{if } \sigma = \epsilon \text{ and } q \in F_1 \\ \phi & \text{if } \sigma = \epsilon \text{ and } q \notin F_1. \end{cases}$$

We intended to use NFA in Figure 1a, but wrongly stated DFA there. So the problem, while doable, is a bit strange. We grade this problem based on if you made some serious mistakes.

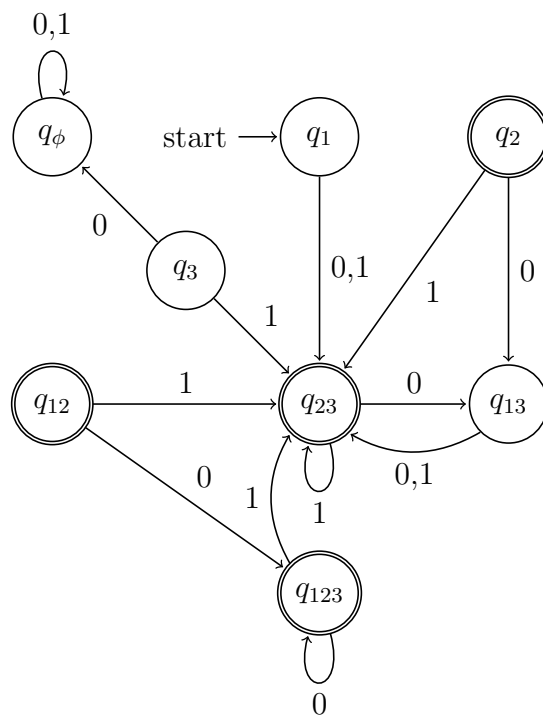
Problem 3 (15 pts)

Consider following NFA:

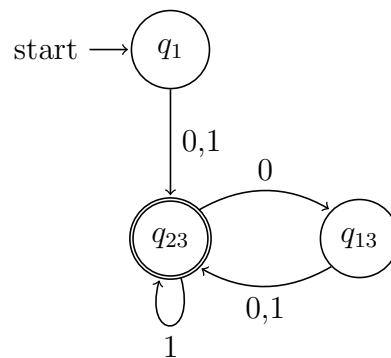


Please convert this NFA to DFA and remove unnecessary state in final answer.

Answer

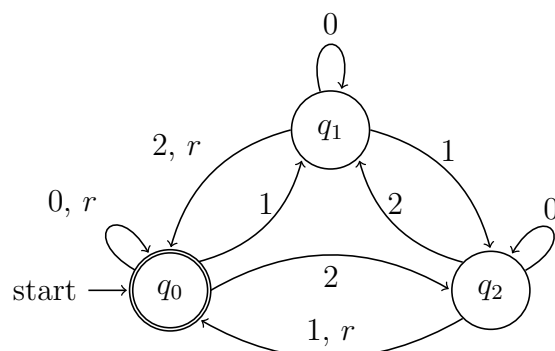


The simplified diagram is



Problem 4 (20 pts)

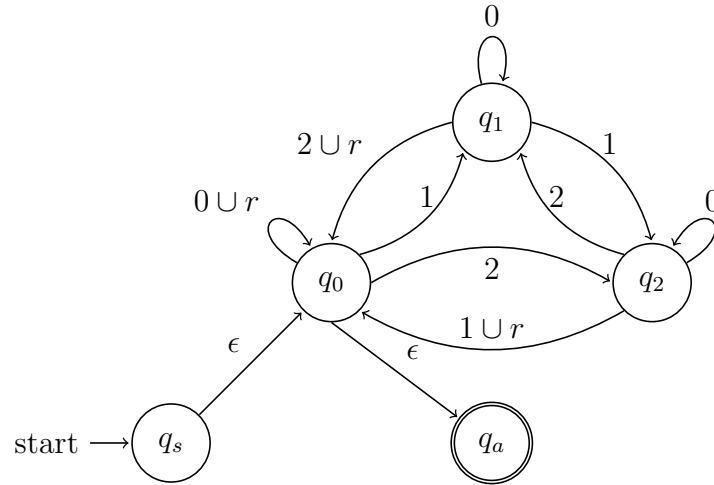
Given DFA in Figure 1.14.



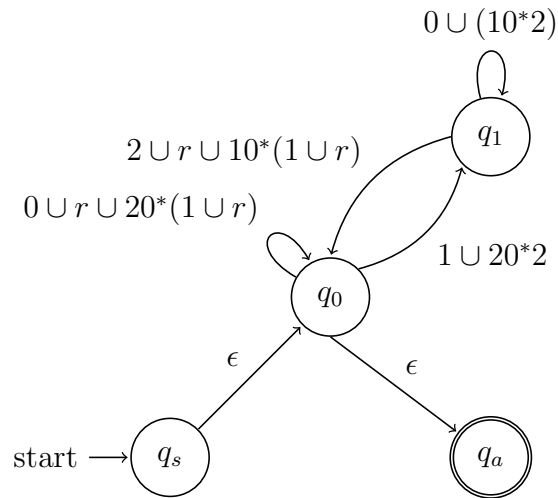
$\Sigma = \{r, 0, 1, 2\}$ and we treat r as a single symbol. Transform it to a GNFA and then obtain a regular expression by sequentially removing state q_2 , q_1 , and q_0 .

Answer

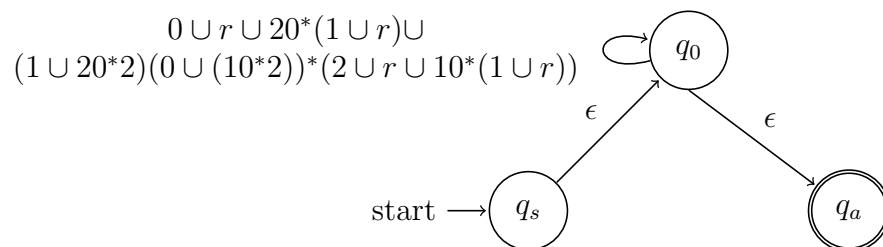
Step 1, add q_s and q_a :



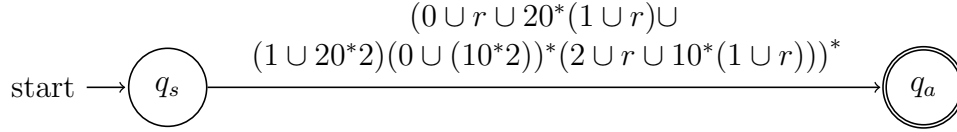
Step 2, remove q_2 :



Step 3, remove q_1 :



Step 4, remove q_0 :



The regular expression is $(0 \cup r \cup 20^*(1 \cup r) \cup (1 \cup 20^*2)(0 \cup (10^*2))^*(2 \cup r \cup 10^*(1 \cup r)))^*$.

Common Mistakes

1. Forget $*$ at the final step.
2. Missing parentheses. For example, someone wrote

$$(0 \cup (10^*2))^*2 \cup r \cup 10^*(1 \cup r)$$

rather than

$$(0 \cup (10^*2))^*(2 \cup r \cup 10^*(1 \cup r)).$$

Problem 5 (15 pts)

Is the following language regular?

$$L = \{0^i1^j \mid \gcd(i, j) = 1\},$$

where $\gcd(i, j)$ means the greatest common divider of i and j . Explain your answer clearly.

Answer

No. Assume the language is regular. By pumping lemma, there is a positive integer $p \geq 1$ such that for all $s \in L$, $s = xyz$ satisfying the following conditions:

- $|y| \geq 1$
- $|xy| \leq p$
- for all $i \geq 0$, $xy^iz \in L$

Let $s = 0^q1^{(q-1)!}$, where q is the smallest prime $\geq p$. Then, $\gcd(q, (q-1)!) = 1$. Since $0^q \geq p$, y must be 0^k for some $k \geq 1$. For $i = 0$, $xy^0z = 0^{(q-k)}1^{(q-1)!}$, but $\gcd(q-k, (q-1)!) = q-k$ if $k \geq 1$. We have proved $xy^0z \notin L$ and got a contradiction, so the language L is not regular.

Problem 6 (20 pts)

Is the following language regular?

$$L = \{uww^Rv \mid u, v, w \in \{0, 1\}^+, |u| \geq |v|\},$$

where $\{0, 1\}^+$ means the set of strings which are composed of 0 and 1 (ϵ is not included). And w^R means the reverse of the string w . Explain your answer clearly.

Answer

No. Assume the language is regular. By pumping lemma, there is a positive integer $p \geq 1$ such that for all $s \in L$, $s = xyz$ satisfying the following conditions:

- $|y| \geq 1$
- $|xy| \leq p$
- for all $i \geq 0$, $xy^iz \in L$

Let $s = (01)^p(10)^p$. We have $s \in L$ because we can choose $w = 1$, $u = (01)^{p-1}0$, and $v = 0(10)^{p-1}$. And the only way to cut s to uw and w^Rv is to cut between $(01)^p$ and $(10)^p$. For this case, $|u| = |v|$, $uw = (01)^p$, $w^Rv = (10)^p$. And because $|(01)^p| \geq p$, y must be some part of $(01)^p$.

Now we claim for $i = 0$, $xy^0z \notin L$. Since $|u| \geq |v|$, $|uw| \geq |w^Rv|$, and there is no way to cut $(10)^p$, the possible cut we can find in xy^0z to let $|uw|$ longest is the cut we found in s . Since we take off some part of $(01)^p$ (the y), $|uw| < |w^Rv|$, and $|u| < |v|$.

We can claim $xy^0z \notin L$ by the above statements and get a contradiction. So the language L is not regular.

Other Solutions

You can choose s to be $(01)^p11(01)^p$ or $1^{2p}0110(10)^p$. They can both be used to obtain a contradiction.

Common Mistakes

1. When you choose a string s , you cannot specify x , y , or z there. For example, you cannot choose s to be $0^{p-1}0^y0^y11^{p-2}$.
2. You can not choose the pumping length.