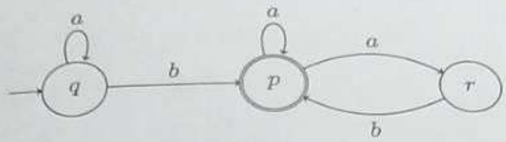
(2 points) Question 1. Consider the following automaton $\mathcal{A}$.



2

(a) Is $\mathcal{A}$ deterministic or non-deterministic?
(b) Is aaa accepted by $\mathcal{A}$?
(c) Is $abababa$ accepted by $\mathcal{A}$?
(d) Construct the deterministic automaton for $\mathcal{A}$.

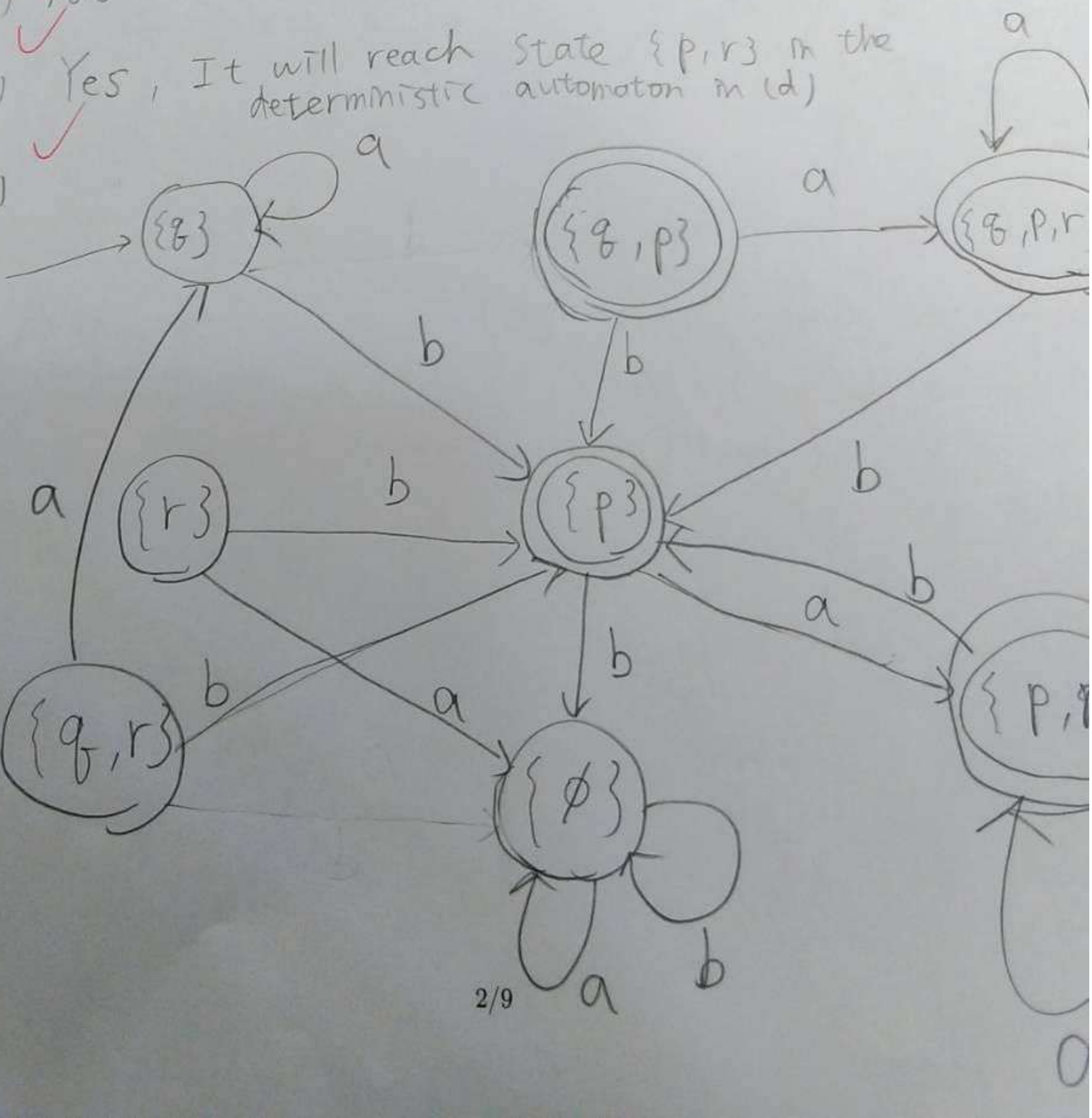
Solutions for Question 1.

(a) It is non-deterministic, since state p has two out edges with input character 'a'

(b) No

(c) Yes, It will reach state {p,r} in the deterministic automoton in (d)

(d)



(2 points) Question 2. Consider the following grammar $\mathcal{G}$ over the alphabet $\{a, b\}$, where S is the start variable.

$$S \rightarrow \epsilon \mid aSb \mid bSa \mid SS$$

2.0

Determine which of the following words are in $L(\mathcal{G})$.

(a) $abab$.

(b) a^3b^3, i.e., $aaabbb$.

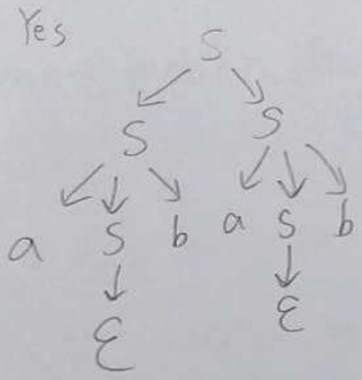
(c) a^4b^3, i.e., $aaaabbb$.

(d) a^2b^3, i.e., $aabbb$.

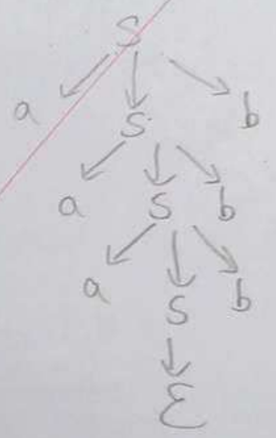
If you think a word is in $L(\mathcal{G})$, you should provide its derivation tree. If you claim it is not, then just state so and you don't need to "prove" it.

Solutions for Question 2.

(a) Yes



(b) Yes



(c) No, since the number of a is not equal to the number of b

(d) No, since the number a is not equal to the number of b

(1 points) **Question 3.** Define a language L with the following property.

- L is not regular.
- There is an integer $N \geqslant 0$ such that every word $w \in L$ with length $\geqslant N$ can be partitioned into three parts:

$$w = xyz$$

where $|y| \geqslant 1$ and for every integer $i \geqslant 0$, $xy^iz \in L$.

1

no prove
simple explane

Solutions for question 3.

Let $L = \{ a^n b^n c^m \mid n, m > 0 \}$

It is obvious that L is not regular, since it doesn't meet the pumping lemma in regular langua

But, there's an integar N, such that $\forall w$ s.

$|w| \geq N$, I can split the string $w = xyz$,

where $w = \underbrace{aa \cdots aabb \cdots bb}_{x}\underbrace{cc \cdots c}_{y}\underbrace{c}_{z}$

since $m > 0 \Rightarrow |y| \geq 1$, and $\forall i$, $xy^iz \in L$,

(1 point) Question 4. Define two languages L_1 and L_2 with the following properties.

$L_1 \neq L_2$

1

- L_1 is CFL, but not regular.
- L_2 is CFL, but not regular.
- $L_1 \cap L_2$ is CFL, but not regular.

For this question, you only need to present the CFG for L_1, L_2 and $L_1 \cap L_2$.

Solution for question 4.

- $L_1 = \{a^n b^n c^m \mid n, m > 0\}$

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

  $\Gamma = \{S, Y, Z\}$

  starting variable $= S$

  $S \to aYbcZ$

  $Y \to aYb \mid \varepsilon$

  $Z \to cZ \mid \varepsilon$

- $L_1 \cap L_2 = \{a^n b^n \mid n > 0\}$

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

  $\Gamma = \{S, Y\}$

  starting variable $= S$

  $S \to aYb$

  $Y \to aYb \mid \varepsilon$

- $L_2 = \{a^n b^n d^m \mid n, m > 0\}$

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

  $\Gamma = \{S, Y, Z\}$, starting variable $= S$

  $S \to aYbdZ$

  $Y \to aYb \mid \varepsilon$

  $Z \to dZ \mid \varepsilon$

(1)

(1 point) Question 5. In the following the alphabet is $\Sigma = \{a, b, c\}$. For a word $w \in \Sigma^*$:

- $\#_a(w)$ denotes the number of a's in w.
- $\#_b(w)$ denotes the number of b's in w.
- $\#_c(w)$ denotes the number of c's in w.

Consider the following language over the alphabet $\Sigma = \{a, b, c\}$:

$$L \ := \ \{w \in \Sigma^* \mid \#_a(w) = \#_b(w) = \#_c(w)\}$$

Prove that L is not CFL.

Solution for question 5.

If L is a CFL and K is regular, then $L \cap K$ is a CFL (proved in the homework already)

Assume that L is an CFL, consider a regular language $K = a^* b^* c^*$, so, $L \cap K = \{a^n b^n c^n \mid n \geq 0\}$ must be a CFL, while $\{a^n b^n c^n \mid n \geq 0\}$ is not a CFL, which is proved in the lecture. Contridiction!

So, L is not an CFL!

(1 point) **Question 6.** A string $w \in \{0,1\}^*$ represents an integer in a standard way. For example, the string 000 represents the integer 0, and so do 0 and 000000. The string 00100 and 100 both represent the integer 4. The empty string ϵ represents the integer 0.

For an integer $n \geqslant 1$, define the language L_n over the alphabet $\{0,1\}$ as follows:

$$L_n := \{w \mid w \text{ represents an integer divisible by } n\}$$

3

Is it true that for every integer $n \geqslant 1$, the language L_n is regular? Justify your answer. If you think L_n is regular, give an upper bound on the number of states needed for a DFA to accept L_n for every integer $n \geqslant 1$?

Minimal

Solution for question 6.

Yes, the language L_n is regular, and the number of states need in a DFA is n. upper bound on

The construction of the DFA of L_n is as follows:

$\Sigma : \{0,1\}$

$Q : \{0,1,2,\ldots,n-1\}$

$q_0 : \{0\}$

$F : \{0\}$

$\delta : \left\{ \begin{array}{l} \delta(i,0) = (2i) \bmod n \\ \delta(i,1) = (2i+1) \bmod n \end{array} \right\}$

The idea is that, I record the current number modulo n as the state, and if I redd '0' as the input, I multiply the number by 2. If I read '1' as the input, I multiply the number by 2, and add the number by 1,

The idea is similar to L_3 in the home work!

Bonus question. For a language $L \subseteq \Sigma^*$, define:

$$\mathsf{SQRT}(L) := \{x \mid \text{there is } y \text{ such that } |y| = |x|^2 \text{ and } xy \in L\}$$

Prove that if L is regular, then $\mathsf{SQRT}(L)$ is also regular.

Solution for bonus question.

Since L is a regular language, then I can use a DFA $A = (\Sigma_L, Q_L, q_{0L}, F_L, \delta_L)$ to represent it.

Define some regular languages as follows:

$A_q = \{S \mid$ If the DFA read s as input, it will terminate at state $q\}$

$B_q = \{S \mid$ starting from state q and read s as input, it will terminate in an accepting state in the DFA$\}$

It is straightforward that A_q, B_q are regular languages, since I can make a DFA based on A, but change accepting states and initial state.

Define a language $C(A, B)$ as follows:

$C(A, B) = \{x \mid x \in A$, and there's $y \in B$, such that $|y| = |x|^2\}$, where A, B are two languages.

If $C(A, B)$ is a regular language, then $\mathsf{SQRT}(L) = \bigcup_{q \in Q} C(A_q, B_q)$ is also a regular language. So now, the only thing I need to prove is that $C(A, B)$ is a regular language