Aufgabe 2

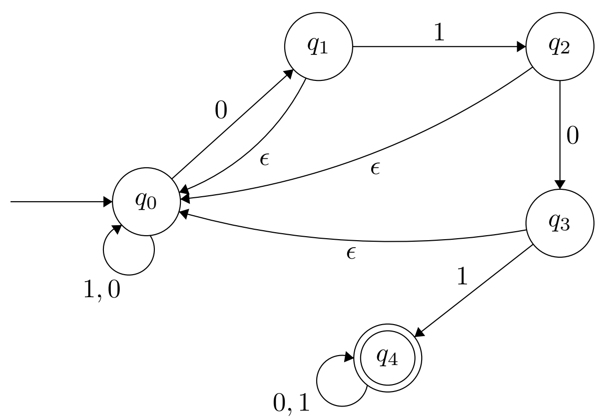
Tim Neutze
5578777
Lorenzo Tecchia
5581906
2023.05.22

Contents

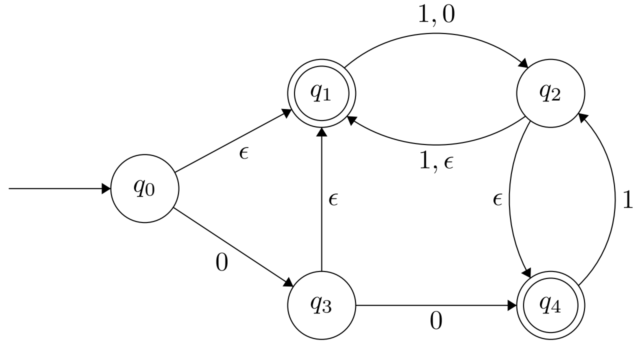
Chapter 1

Task 1

1.1 c)



1.2 d)



Chapter 2

Task2

2.1 a)

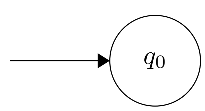
If we chose the machine M as follows:

$$M = (q_0, \emptyset, \delta, q_0, q_0)$$

Then it can be proved that the only language accepted by a machine which has only the start state is the empty string ϵ. So the second machine would have as complement language

$$\Sigma^* \backslash \epsilon \to \epsilon \backslash \epsilon = \emptyset$$

remembering that $\emptyset^* = \epsilon$.
So this would mean that the machine M' would be something similar to this.



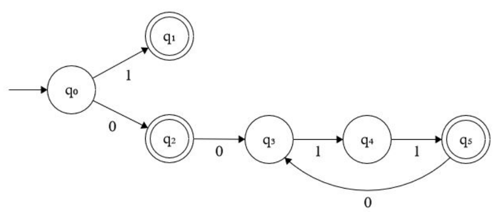
Thus meaning that no language could be accepted by this machine. Since also M' has like acceptance state $Q \backslash F \to q_0 \backslash q_0 \to \emptyset$.

2.2 b)

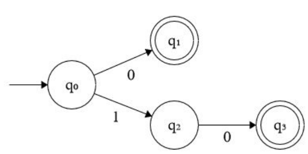
To disprove the statement, it is enough to show a counterexample. Suppose L contains at least two distinct words w_1 and w_2, where w_1 is not equal to w_2. Since M accepts language L, there are paths in M that accept w_1 and w_2. Since M is a NFA, there can be different paths that lead to the same accepting states. Now suppose that M' is an NEA that has exactly one accepting state and $L(M') = L$. Since L contains at least w_1 and w_2, M' must have paths that accept w_1 and w_2. Since M' has only one accepting state, these paths must lead to that state. Since w_1 and w_2 are different words, there is a difference in the paths that accept them. This means that M' is not able to distinguish w_1 and

w_2 and thus is not able to accept L correctly. Therefore, there cannot be an NEA M' with exactly one accepting state such that $L(M') = L$. An example for this is a machine that accepts only the word 0 and the word 10. For this there is no machine with only one state.

2.3 c)



(a) L1

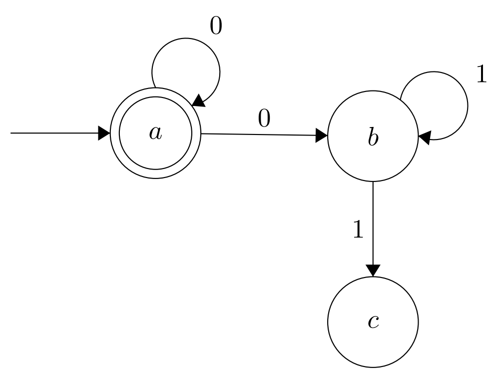


(b) L2

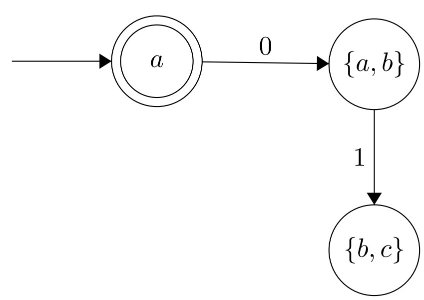
Chapter 3

Task 3

3.1 a)



3.2 b)



Chapter 4

Task 5

4.1 b)

Given the language $L = \{1^{2^n} | n \geq 0\}$ via $\Sigma = \{0, 1\}$ we use the pumping lemma. If we took the pumping length of 1, the string of the language L would be containing 11(also the string having a length more than p). In this case though choosing the substrings xy^iz would mean we choosing the substrings $x = \epsilon$ or $z = \epsilon$. So then taking the substring y and repeating it an arbitrary amount of times, say 2 would mean that the substring containing $xyyz$ would still be accepted by the language, but it's clear that a string containing 111 wouldn't be accepted by the language since the only string of length odd that the language accepts is the string containing just 1. And that is not the case.
Also the language is basically $11^p \rightarrow p > 0$ which means that the length will aways be bigger than the pumping length which means that the pumping Lemma doesn't apply so the language is not regular.

4.2 f)

Given the language $L = \{0^n 1^m 0^{n+m} | n, m \in \mathbb{N}\}$ using the same principles stated above we can choose arbitrarily the substring y. If we chose any number of 1 followed by the last element of the string being a 0 and repeating this string any amount of time, we would then see an alternating amount of substrings of 01 that would ultimately not be accepted by the language since, it's defined in a way that are chunks of 0 followed by chunks of 1 and again chunks of 0 but not the numbers mixed in between. So the language is not regular.