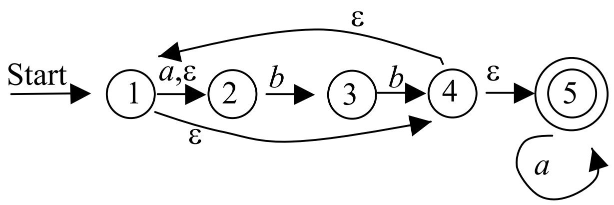
Exercises for FLL, Fall 2018, sheet 3 – Solutions

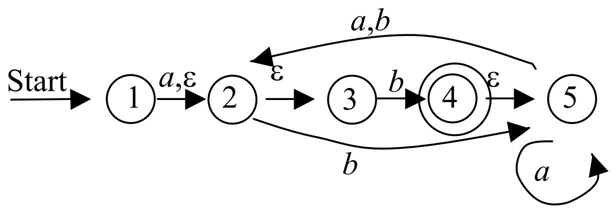
Return Thu Oct 4, in class. As always in this course you may work in teams of two if you wish – but not larger teams.

Exercise 1. Design an ε-NFA that accepts the language denoted by (((ε+*a*)*bb*)*)*a**. Represent your automaton by a transition diagram.

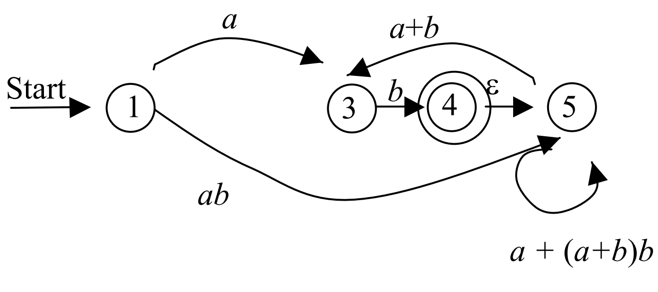
Solution. The diagram shows a possible ε-NFA for our language.



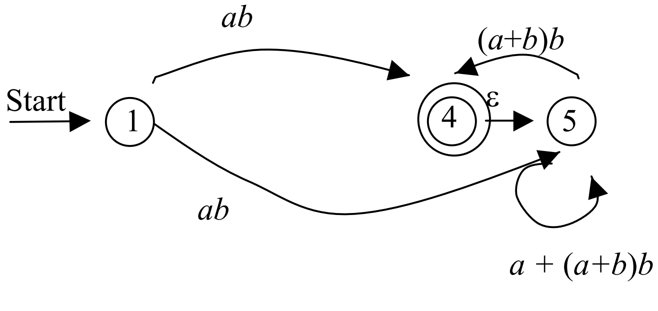
Exercise 2. Give a regular expression which describes the language accepted by the following ε-NFA:



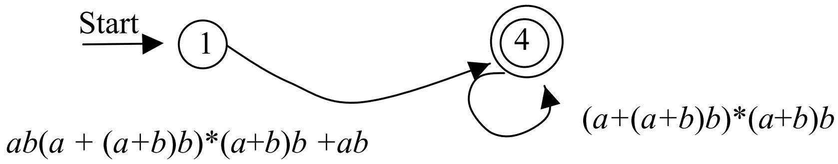
Solution: Using the procedure from the LN, in step 1 we merge away state 2 (and re-write every link label as a regexp, saving some parentheses and some redundant ε), giving



In step 2 we get rid of state 3, yielding



Then we merge away state 5:



This gives the final regexp

$(ab(a + (a+b)b)^*(a+b)b +ab)(a+(a+b)b)^*(a+b)b$

which is written here using some parentheses omitted. Note 1: this regexp can be further simplified a little. Note 2: if you choose another order of deleting states, you will get a different-looking regexp.

Exercise 3. Is the language $L = \{w \in \{0, 1\}^* \mid w$ contains an equal number of 0's and 1's$\}$ regular? Prove your answer.

Solution. Language is not regular. Proof via PL. Assume it is regular and PL holds. Let n be PL constant. Then $w = 0^n1^n \in L$. But according to PL, some nonempty subword of the first part $y = 0^m$ exists such that also $0^{n+m}1^n \in L$, which it isn't , contradiction, thus L no regular.s

Exercise 4 [optional]. Prove that the language $L = \{0^n \mid n = pq$ for two primes $p, q\}$ is not regular.

Solution. Assume L is regular. Then by the pumping lemma, there exists a constant c such that, if $w \in L$, $|w| > c$, w can be written as uvx, $r = |v| > 0$, such that $uv^ix \in L$ for all $i \geq 0$. Let p, q such that $n = pq > c$. Then $0^{pq} \in L$ and by the pumping lemma, $0^{pq+ir} \in L$ for all $i \geq 0$. Specifically, for $i = pq$ we obtain $0^{pq(1+r)} \in L$. But $pq(1+r)$ is not the product of two primes, so by the definition of L, $0^{pq(1+r)} \notin L$. Contradiction, therefore the assumption that L is regular is wrong, therefore L is not regular.

Exercise 5. Prove the following claim:

Let M be some regular language over $\Sigma = \{0,1\}$. Define $L_{|M|} = \{0^n \in \{0\}^* \mid n = |v|$ for some word $v \in M\}$. Then L is regular.

Note: this is easy to solve using the tool of language homomorphisms – leads to a one-line proof. It is also possible to solve this problem without homomorphisms, also not difficult, though a bit more lengthy.

Solution. The conjecture is true. Proof with homomorphisms: Consider the (unique!) homomorphism h: $\{0,1\} \to \{0\}$. Then $L_{|M|} = h(M)$ and by Proposition 3.11, $L_{|M|}$ is regular. Proof without homomorphims: Let A be a DFA for M. Relabel all "1"-transitions in A by 0, obtain an NFA A'. This NFA accepts $L_{|M|}$. To see this, first consider some $v \in M$. It has a path in A that ends in an accepting state; the same path in A' leads the input word $0^{|v|}$ to the same accepting state. This if $n = |v|$ for some word $v \in M$, $0^n \in L(A')$, that is, $L_{|M|} \subseteq L(A')$.

Conversely, let $0^n \in L(A')$. Then there is a path of 0-transitions of length n in A' that leads from the starting state to some accepting state. If you replace the 0's on this path that were obtained from 1 -> 0 relabelings by the original 1's, connecting the {0, 1} transitions on this path you get a word $v \in M$, $n = |v|$. Hence $L(A') \subseteq L_{|M|}$. Therefore, $L(A') = L_{|M|}$ and $L_{|M|}$ is regular.

Exercise 6 A *sequence language* L over Σ is a language with two properties: (i) for each $n \geq 0$, there exists exactly one word in L of that length; (ii) if $u, v \in L$, $|u| < |v|$, then u is a prefix (= initial subword) of v.

a. Give an example of a regular sequence language.

b. Prove that every regular sequence language L is ultimately cyclic, that is, there exist words w and v such that L is the set of all initial substrings of the infinite sequence *wvvvv...* . *Hint*: you will benefit from the PL here.

Note. This was a midterm question in a long-time-ago FLL course.

Solution. a. The simplest example surely is the language $L(\mathbf{1}^*)$.

b. Let L be a regular sequence language. Because L is regular, the PL holds for L. Let n be a PL constant for L. Because a sequence language is infinite, there must exist some w in L with $|w| \geq n$. By the PL, w can be split into $w = xyz$, where $|y| > 0$, $|xy| \leq n$, and every xy^kz (where $k \geq 0$) is in L. Thus L contains all the words xz, xyz, $xyyz$, $xyyyz$, etc. Because in a sequence language L it holds that if $u \in L$, $v \in L$, $|v| < |u|$, then $u = vx$ for some x, we can conclude from $xy^kz \in L$ that $x, xy, xyy, ..., xy^k \in L$. Because $xy^kz \in L$ for all k, we conclude that $xy^k \in L$ for all k. By the fact that sequence languages are closed under initial subwords, it follows that all initial strings of the infinite sequence *xyyy...* are in L. Because L is a sequence language, no other words can be in L. Thus L is the language of all initial substrings of *xyyy...*, hence L is ultimately cyclic.