NTIN071 A&G: Tutorial 5 – Regular expressions (bonus: 2-way automata)

Teaching goals: The student is able to

- construct a regular expression for a language given in set notation
- convert a regular expression to a finite automaton
- convert a finite automaton to a regular expression

In-class problems

Problem 1 (Constructing regular expressions)**.** Find regular expressions representing the following languages over $\Sigma = \{a, b\}$ consisting of words that:

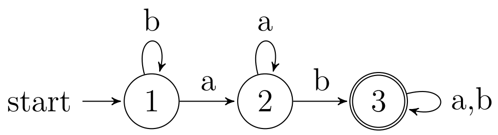
(a) start with 'abba',
(b) start with 'ab' and end with 'ba',
(c) contain 'abba' or 'bab' as a subword,
(d) do not contain 'aa' as a subword,
(e) contain an even number of a's,
(f) the first letter is the same as the last.

Problem 2 (Regex to automaton)**.** Construct NFAs recognizing the languages described by the following regular expressions:

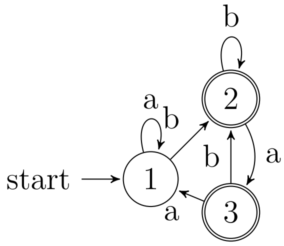
(a) $a^2 + b^2 + ab$
(b) $a + b^*$
(c) $(ab + c)^*$

Problem 3 (Automaton to regex)**.** Construct regular expressions for languages recognized by the following automata.

(a)



(b)



Problem 4 (Complement of a Regular Expression)**.** Consider the following regular expression over the alphabet $\Sigma = \{a, b\}$ and let $L = L(R)$.

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

(a) Construct a *nondeterministic* finite automaton A (as small as possible) recognizing L.
(b) Use the subset construction to convert A to a *deterministic* finite automaton B.
(c) From the automaton B, construct a DFA C recognizing the *complement* of L.

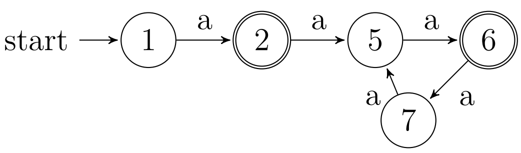
Extra Practice and Thinking

Problem 5 (Regex to automaton)**.** Construct finite automata accepting languages described by the following regular expressions:

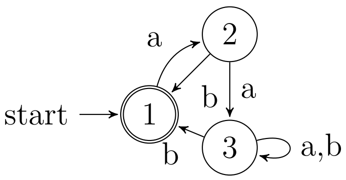
(a) $ab + ba$
(b) $((ab + c) + a(bc)^* + b)^*$
(c) $((ab + c)^* a(bc)^* + b)^*$
(d) $(01^* + 101)^* 0^* 1$

Problem 6 (Automaton to regex)**.** Construct regular expressions for languages accepted by the following automata.

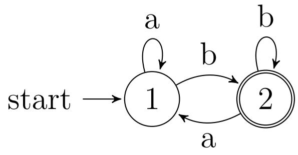
(a)



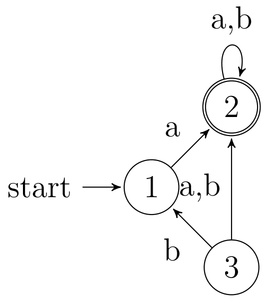
(b)



(c)



(d)



Problem 7 (Testing equivalence of regular expressions)**.** Describe an algorithm to test equivalence of two regular expressions. Apply it to $(a+b)(a+b)^*$ and $a(a+b)^*+b(a+b)^*$.

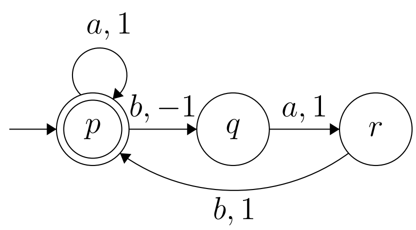
Problem 8 (Are regular expressions regular?)**.** Fix a finite alphabet Σ. Is the language consisting of all regular expressions over Σ a regular language?

Bonus: Two-way automata

Problem 9 (Convert a 2-way automaton)**.** Consider the following two-way DFA.

(a) Determine the language it recognizes.
(b) Determine the functions f_u and the congruence $\sim$ for all words of length at most 4.
(c) Convert it to an equivalent one-way automaton.

	a	b
$\to *p$	$p, 1$	$q, -1$
q	$r, 1$	
r		$p, 1$



Problem 10 (Without 2-way automata this is hard)**.** Given a DFA A, design an NFA recognizing the language $L' = \{\#w\# \mid ww^R \in L(A)\}$. ((Do not use two-way DFAs.)

Problem 11 (Constructing 2-way automata)**.** Let L be a regular language over Σ and $\# \notin \Sigma$. Construct a two-way finite automaton accepting the given language:
(a) $L' = \{\#w\# \mid ww^R \in L\}$
(b) $L' = \{\#w\# \mid (\exists u \in \Sigma^*)(wu \in L \ \wedge \ |w| = |u|)\}$