Mälardalen University
School of Innovation, Design and Engineering

TEN1 DVA337 2016

Formal languages, automata, and theory of computation

Thursday, January 7, 14:10 - 18:30
Teacher: Daniel Hedin, phone 021-107052 (15:00 - 17:00)

The exam has a total of 40 points distributed over 14 questions. No aids are allowed. Answers must be given in English and should be clearly justified.

- Explain all solutions. A correctly explained solution with minor mistakes may render full points.
- Write clearly and *use the definitions and notation we have used in the course*. Unreadable solutions will not be graded.
- Start each question on a new page and only write on one side of the page.

Regular languages (14 p) The NFA in Figure 1 accepts a language L over $\Sigma = \{a, b\}$.

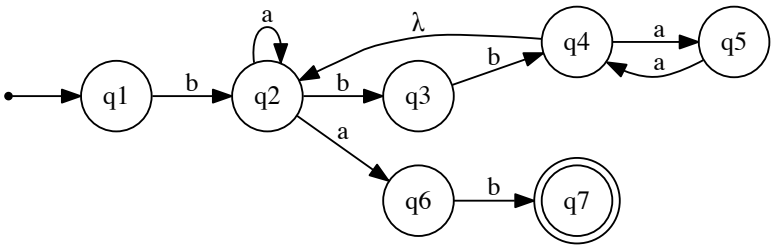


Figure 1: An NFA

1) Construct a regular expression for L. (2p)

Solution:

$$b(a|bb(aa)^*)^*ab = b(a|bb)^*ab$$

---

2) Construct a regular grammar for L. (2p)

Solution: Corresponding to the first regular expression

$$\begin{array}{lcl} S & \rightarrow & bA \\ A & \rightarrow & aA \mid bbB \mid ab \\ B & \rightarrow & aaB \mid A \end{array}$$

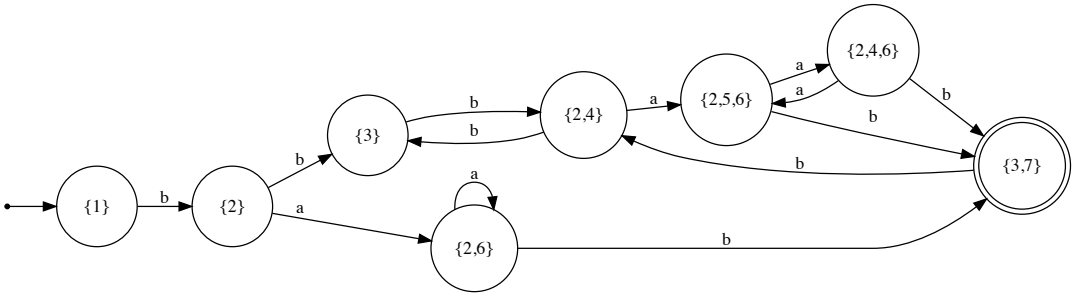
Corresponding to the second regular expression

$$\begin{array}{lcl} S & \rightarrow & bA \\ A & \rightarrow & aA \mid bbA \mid ab \end{array}$$

---

3) Construct a DFA for L. Note that the DFA does not have to be minimal. (3p)

Solution: Proceed by subset construction.



---

4) Minimize the DFA in Figure 2. (3p)

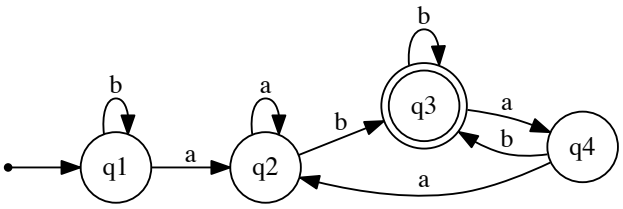
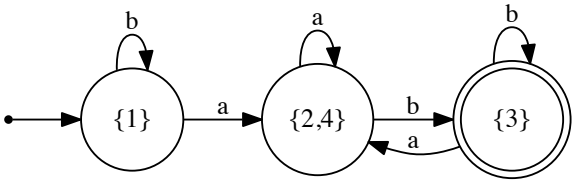


Figure 2: A DFA

Solution: Proceed with set partitioning starting with two sets $\{1,2,4\}$ and $\{3\}$.



---

5) Use the pumping lemma to prove that $L = \{a^p b^q \mid p = k * q \text{ for } p,q,k > 0\}$ is not regular. A detailed proof is needed for full points. (4p)

Solution: We proceed with a proof by contradiction. Assume that L is regular. The pumping lemma now states that all strings w such that $|w| \geq m$ for some cut-off m has *at least one decomposition* $xyz = w$ subject to the constraints that $|xy| \leq m$ and $|y| \geq 1$ where y can be pumped, i.e., $\forall i \; . \; xy^i z \in L$.

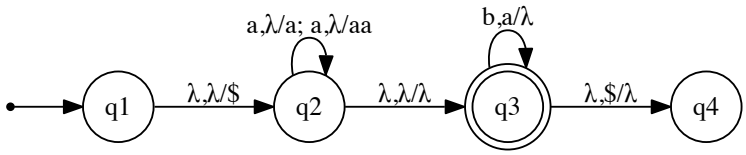
We will derive a contradiction by showing the existence of a string in the language that cannot be pumped, i.e., that for at least one i, $xy^i z \notin L$. Consider $w = a^m b^m$. Clearly, $w \in L$ (for $k = 1$; the language consists of the string produced for all $k > 0$) and $|w| = 2 * m \geq m$. Now, due to the constraint that $|xy| \leq m$ and $|y| \geq 1$ we have that *all* decompositions will satisfy that $y = a^l$ for some l, where $m \geq l \geq 1$.

Consider $xy^0 z = a^{m-l} b^m$ (by removing y we remove a^l from the string. For this to be in the language we would need $m - l = k * m$ for some $k > 0$, but this is not possible for l such that $m \geq l \geq 1$. Thus $xy^0 z \notin L$ which contradicts the pumping lemma that at least one of the decompositions should allow *all* pump indexes.

Context-free languages (14 p)

6) Construct a push-down automaton for the language $L = \{a^n b^m \mid n \leq m \leq 2n\}$. (3p)

Solution:



---

7) What language does the NPDA in Figure 3 accept? (3p)

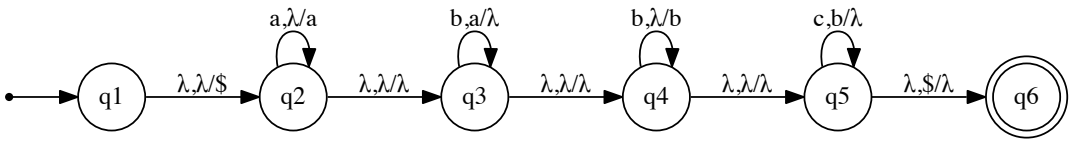


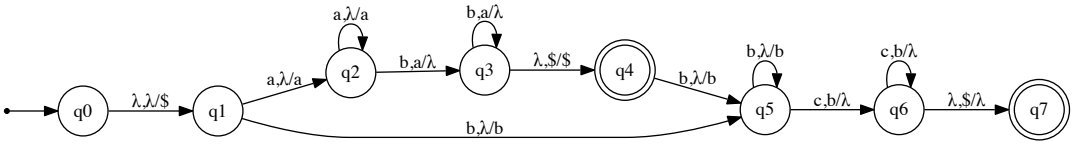
Figure 3: An NPDA

Solution: $\{a^i b^j c^k \mid i + k = j\}$

---

8) Construct a DPDA that is equivalent to the NPDA in Figure 3. (3p)

Solution:



---

9) Construct a context-free grammar for the language $L = \{a^n b^n c^m d^m \mid m,n \geq 0\}$. (2p)

Solution:

$$\begin{array}{lcl} S & \rightarrow & AB \\ A & \rightarrow & aAb \mid \lambda \\ B & \rightarrow & cBd \mid \lambda \end{array}$$

---

10) Explain the notion of ambiguity and show that the following grammar is ambiguous. (3p)

$$\begin{array}{lcl} S & \rightarrow & Ab \mid aaB \\ A & \rightarrow & a \mid Aa \\ B & \rightarrow & b \end{array}$$

Solution: A grammar is ambiguous if there exists (equivalently)

- two left derivations,
- two right derivations, or
- two derivation trees

resulting in the same string. For the above grammar there exists two left derivations of the string *aab*

1. $S \Rightarrow aaB \Rightarrow aab$
2. $S \Rightarrow Ab \Rightarrow Aab \Rightarrow aab$

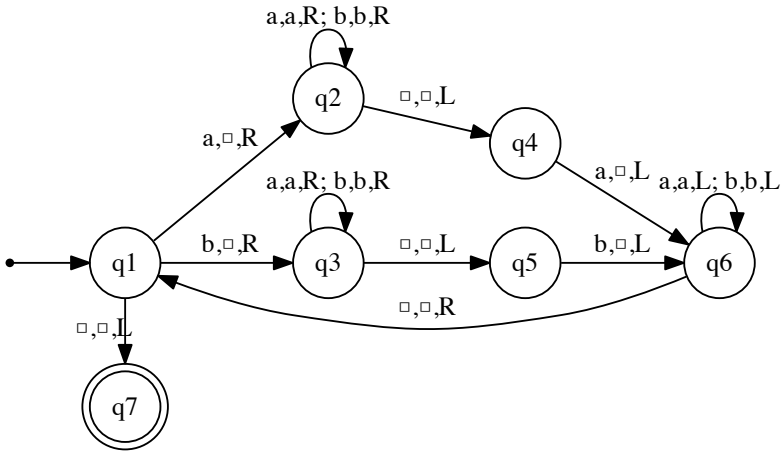


Figure 4: Turing machine M

Restriction-free languages and theory of computation (12 p) Consider the Turing machine M in Figure 4.

11) Write down δ, the transition function of M. (2p)

Solution:

$$M = (\{q_1, q_2, q_2, q_3, q_4, q_5, q_6, q_7\}, \{a, b\}, \{a, b, \Box\}, \delta, q_1, \Box, \{q_7\})$$

with δ defined as follows

$$\begin{array}{lcllcllcl}
\delta(q_1,a) & = & (q_2,\Box,R) & \delta(q_2,a) & = & (q_2,a,R) & \delta(q_3,a) & = & (q_3,a,R) \\
\delta(q_1,b) & = & (q_3,\Box,R) & \delta(q_2,b) & = & (q_2,b,R) & \delta(q_3,b) & = & (q_3,b,R) \\
\delta(q_1,\Box) & = & (q_7,\Box,L) & \delta(q_2,\Box) & = & (q_4,\Box,L) & \delta(q_3,\Box) & = & (q_5,\Box,L) \\
 & & & & & & & & \\
\delta(q_4,a) & = & (q_6,\Box,L) & \delta(q_5,b) & = & (q_6,\Box,L) & \delta(q_6,a) & = & (q_6,a,L) \\
 & & & & & & \delta(q_6,b) & = & (q_6,b,L) \\
 & & & & & & \delta(q_6,\Box) & = & (q_1,\Box,R)
\end{array}$$

---

12) What language does M accept? (2p)

Solution: M recognizes the language of even length palindromes, $\{ww^R \mid w \in \Sigma^*\}$. To see this consider how M operates. First, M assume that M is started with some input w, i.e., corresponding to the instantaneous description q_1w. First, the empty string is accepted, since if the first character is the blank symbol $\Box$, M accepts. Otherwise, the first symbol is either an a, or a b. In both cases, M overwrites it with the $\Box$ and searches for the last non-blank symbol if it exists. If it does not exist or if it does not match the overwritten symbol the machine halts. Otherwise, M overwrites the matching symbol and searches for the first non-blank symbol. If it exists, the process repeat until no more symbols remain. At this point M accepts.

Phrased more distinctly, M shaves off matching symbols in both ends until only the empty string remains, which is accepted.

---

13) Select a string $w_a \in \{a,b\}^4$ that is accepted by M and a string $w_r \in \{a,b\}^4$ and show the executions of M on the selected strings as two sequences of instantaneous descriptions. (4p)

Solution: For $w_a = abba$ we have the following derivation

$\Box\, q_1\, abba\Box \vdash \Box\Box\, q_2\, bba\Box \vdash \Box\Box b\, q_2\, ba\Box \vdash \Box\Box bb\, q_2\, a\Box \vdash \Box\Box bba\, q_2\Box \vdash$
$\vdash \Box\Box bb\, q_4\, a\Box \vdash \Box\Box b\, q_6\, b\Box\Box \vdash \Box\Box\, q_6\, bb\Box\Box \vdash \Box\, q_6\, \Box bb\Box\Box \vdash \Box\Box\, q_1\, bb\Box\Box \vdash$
$\vdash \Box\Box\Box\, q_3\, b\Box\Box \vdash \Box\Box\Box b\, q_3\, \Box\Box \vdash \Box\Box\Box\, q_5\, b\Box\Box \vdash$
$\vdash \Box\Box\, q_6\, \Box\Box\Box\Box \vdash \Box\Box\Box\, q_1\, \Box\Box\Box \vdash \Box\Box\, q_7\, \Box\Box\Box\Box$, halt and accept

For $w_r = abbb$ we have the following derivation

$\Box\, q_1\, abbb\Box \vdash \Box\Box\, q_2\, bbb\Box \vdash \Box\Box b\, q_2\, bb\Box \vdash \Box\Box bb\, q_2\, b\Box \vdash \Box\Box bbb\, q_2\Box \vdash$
$\vdash \Box\Box bb\, q_4\, b\Box$ halt and reject

---

14) Explain the concept of reduction proof. In particular, explain what is reduced to what (be careful with the direction of the reduction) and which conclusions that can be drawn from such reductions. (4p)

Solution: A proof by reduction is a proof where we reduce a known undecidable problem A to another problem B. The reduction is performed by showing how to use a decision procedure for B as a decision procedure for A. Note that the reduction itself has to be computable. This means that the existence of decision procedures for B implies the existence of decision procedure for A. However, since we know that there cannot exist any decision procedures for problem A we draw the conclusion that there cannot exist any decision procedures for B.
