Examination
Formal Languages and Automata Theory
TDDD14

(Formella Språk och Automatateori)

2013–01–08, 08.00 – 12.00

1. Allowed help materials
   - A sheet of notes - 2 sided A5 or 1 sided A4.
     The contents is up to you.
     The notes should be signed in the same way as the exam sheets and returned together with the exam.
   - English dictionary

   (Tillåtna hjälpmedel:
   - Ett papper med valfria anteckningar - 2 sidor A5 eller 1 sida A4. Anteckningarna ska signeras på samma sätt som tentamensarken och bifogas tentamen vid inlämnandet.
   - Engelsk ordbok)

2. You may answer in Swedish or English.
3. Total number of credits is 30: 3: 15 p, 4: 20 p, 5: 25 p.
4. *Jour* (person on duty): Jonas Wallgren, tel. (013 28) 26 82

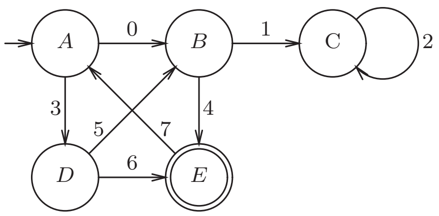
GOOD LUCK !

Make sure that you justify your answers! Unexplained answers will be granted 0 points. (For example, assume that you are writing a grammar for a given language. Then you should also explain that the grammar indeed generates the language).

1. *(2p)* Consider the NFAϵ whose transition function is given by the table. (Its set of states is $Q = \{A, B, C, D\}$, the input alphabet $\Sigma = \{0, 1\}$, the start state is A and the final state is D.) Using a standard method construct an equivalent DFA.

	ϵ	0	1
$\rightarrow A$	$\{D\}$	$\{B\}$	$\emptyset$
B	$\emptyset$	$\{C\}$	$\{A\}$
C	$\{B\}$	$\{D\}$	$\emptyset$
D **F**	$\emptyset$	$\emptyset$	$\{C\}$

2. *(2p)*

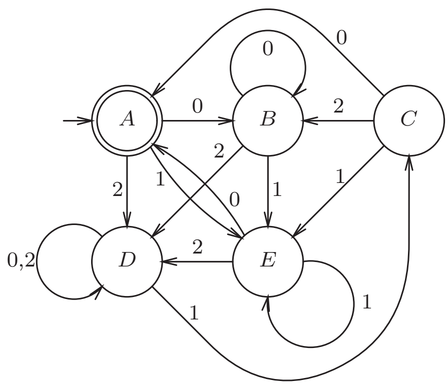


Using a standard method, construct a regular expression defining the same language as the NFA given by the diagram.

(Its set of states is $Q = \{A, B, C, D, E\}$, the input alphabet $\Sigma = \{0, 1, 2, 3, 4, 5, 6, 7\}$, the start state is A, and E is the only final state.)

3. *(2p)* Using a standard method, construct the minimal DFA equivalent to the DFA given by the diagram.

(Its set of states is $Q = \{A, B, C, D, E\}$, the input alphabet $\Sigma = \{0, 1, 2\}$, the start and the final state is A,)



4. *(2p)* Let L be the language accepted by the DFA from the previous problem. Consider the relation $\equiv_L$ on strings over $\Sigma = \{0, 1, 2\}$, defined by

$$x \equiv_L y \iff \forall z \in \Sigma^* \, (xz \in L \Leftrightarrow yz \in L).$$

How many equivalence classes does $\equiv_L$ have? Why? Choose one of the equivalence classes, and give a DFA defining it.

5. *(1p)* Write regular expressions for the languages

$$L_1 = \{\, a^i b^j \mid i+j \text{ is even} \,\},$$
$$L_2 = \{\, x c^i x \mid x \in \{a,b\}^*,\ |x| = 2,\ i \text{ is even} \,\}.$$

6. *(6p)* For each of the following languages answer whether it is regular, context-free but not regular, or not context-free. A brief, informal explanation is sufficient.

   Remember that $\#a(w)$ denotes the number of occurrences of symbol a in w.

   (a) $L_3 = \left\{ w \in \{a,b,c,d\}^* \;\middle|\; \begin{array}{l} w \text{ does not have a substring } aba, \\ \text{each } a \text{ in } w \text{ is immediately followed} \\ \text{by } b,\ \#c(w) \text{ is odd,} \end{array} \right\}.$

   (b) $L_4 = \{\, a^i b^j c^k a^i b^l \mid j > l,\ i,l,k > 0 \,\}$.

   (c) L_5 is the image of L_4 under the homomorphism $h\colon \{a,b,c,d\}^* \to \{0,1,2\}^*$ such that $h(a) = h(b) = 10,\ h(c) = 210, h(d) = \epsilon$.

   (d) L_6 is the image of L_4 under the homomorphism $h\colon \{a,b,c,d\}^* \to \{0,1,2\}^*$ such that $h(a) = h(b) = 210,\ h(c) = h(d) = \epsilon$.

   (e) $L_7 = \{\, a^{i+j} b^{l+k} c^m a^{l+m} b^{i+k} d^j \mid i,j,k,l,m \geq 0 \,\}$.

7. *(3p)* Prove that a language

$$L_8 = \{\, a^i b^j c a^l b^j \mid i < l,\ i,j > 0 \,\}$$

   is not regular, <u>or</u> that it is not context-free. Use the appropriate pumping lemma or employ reasoning similar to the proof of the lemma.

8. *(1p)* Explain briefly the notion of a universal Turing machine (UTM). What is its input alphabet, and the language it accepts?

9. *(3p)* Show that the membership problem is undecidable (i.e. whether a Turing machine M accepts a string x). Use the fact that the problem "a Turing machine M accepts the empty string" is undecidable.

   In other words, show that the language of the problem

$$MP = \{\, \langle M, x \rangle \mid \text{Turing machine } M \text{ accepts input } x \,\}$$

   is not recursive, using the fact that the language

$$EP = \{\, \langle M \rangle \mid \text{Turing machine } M \text{ accepts } \epsilon \,\}.$$

   is not recursive.

10. *(3p)* Which of the following statements are true, which are false? Justify your answers.

    (a) Let us call an NFA $M = (Q, \Sigma, \Delta, S, F)$ "almost deterministic" if $\Delta(p, a)$ has at most one element, for any state $p \in Q$ and symbol $a \in \Sigma$. (So for any given state and input symbol there is at most one next state.)

    If NFA M' is obtained from an almost deterministic NFA M by swapping its final and non-final states, $M = (Q, \Sigma, \Delta, S, F)$ and $M' = (Q, \Sigma, \Delta, S, Q \setminus F)$, then $L(M') = \Sigma^* \setminus L(M)$.

    (b) There exists a regular language for which no LR(0) grammar exists.

    (c) If languages L_1, L_2 are recursively enumerable then $L_1^R \cap L_2$ is recursive.

    Remember that L^R is the language of the reversed strings from L, $L = \{ x^R \mid x \in L \}$.

11. *(2p)* Construct a grammar which is is not LL(1), has two nonterminal and three terminal symbols, has no useless symbols, and generates an infinite language.

12. *(3p)* In an attempt to construct LR parsers for certain grammars, we applied the standard method of constructing a DFA for the viable prefixes of a grammar. Some fragments of the obtained DFA's are given below.

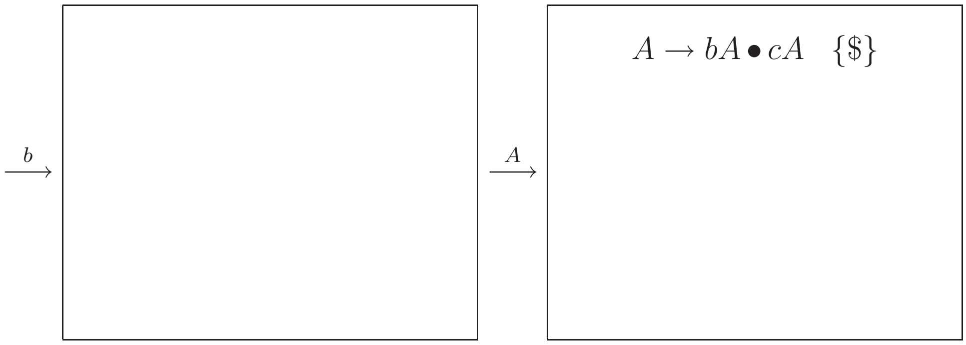
    Complete the missing items in the given states, the missing lookahead sets and the missing symbols labelling the arrows. In each case answer the following questions. Justify your answers.

    - Does the fragment of a DFA satisfy the conditions for the grammar to be LR(0)?
    - The same question about the conditions for LR(1).

    You may skip adding missing items or lookahead sets if they are not needed to answer the questions. For instance if you find the items in some state to violate the LR(1) conditions then you do not need to complete the other states.

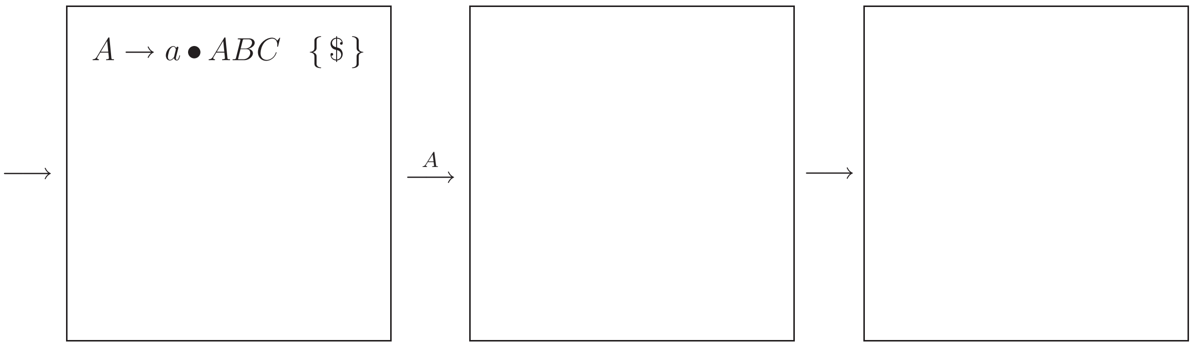
    a, b, c are the terminal symbols and S, A, B, C are the nonterminal symbols of the grammars; S is the start symbol.

(a)



In this grammar the productions beginning with A are $A \to \epsilon \mid bAcA \mid cAbA$.

(b)



The productions of the grammar are $S \to A,\ A \to aABC \mid CB$, $B \to aB \mid C,\ C \to b$.