Exam Languages and Machines
18 June 2015, 14.00-17.00

Duration 3 hours. Closed book. You are allowed to use theorems from the Lecture Notes, provided you phrase them correctly. Give clear and crisp arguments for all your assertions. Write legible.

Write your name on all pages you turn in; number them.

Exercise 1 (10 %). Consider a language L over the alphabet Σ. Fill in the dots (...) with a property of machines defined in the course.

(a) L is context-free $\equiv$ $\exists\, M : L = L(M)$ and M is a ...
(b) L is decidable $\equiv$ $\exists\, M : L = L(M)$ and M is a ...
(c) L is semi-decidable $\equiv$ $\exists\, M : L = L(M)$ and M is a ...
(d) L is regular $\equiv$ $\exists\, M : L = L(M)$ and M is a ...

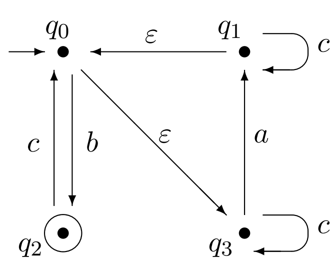
(e) Give all valid implications between these four assertions about L.

Exercise 2 (12 %). Let $G = (V, \Sigma, P, S)$ be a context-free grammar. When is G *essentially noncontracting*? When is G *productive*? Give the two definitions.
(b) Let the context-free grammar G be given by $\Sigma = \{a, b, c\}$, $V = \{S, D, E\}$, and the production rules:

$$\begin{array}{lcl} S & \to & cE \mid aDb \\ D & \to & Sc \mid \varepsilon \mid aE \\ E & \to & bE \mid DD \ . \end{array}$$

Use the standard algorithm to determine an equivalent *productive* grammar. Give and prove all intermediate results.

Exercise 3 (10 %). Consider the alphabet $\Sigma = \{a, b, c\}$ and the nondeterministic finite state machine M with ε-transitions, with the state diagram:



Use the standard algorithm to determine the transition table of an equivalent deterministic finite state machine. Indicate the start state and the accepting states.

Exercise 4 (12 %). (a) Phrase the Pumping Lemma for *regular* languages.
(b) Given is the language $L_4 = \{ww \mid w \in \Sigma^*\}$ over the alphabet $\Sigma = \{a, b\}$. Prove that this language is not regular.

Exercise 5 (11 %). Consider the language L_5 over $\Sigma = \{a, b, c\}$ given by

$$L_5 = \{w \in \Sigma^* \mid n_b(w) \leq 1 + n_c(w)\} \ .$$

Construct a simple pushdown machine M_5 that accepts the language L_5. Give the state diagram, and give convincing arguments that the language accepted by M_5 indeed equals L_5.

Exercise 6 (11 %). Consider the alphabet $\Sigma = \{a, b, c\}$ and the language

$$L_6 = \{w \in \Sigma^* \mid n_a(w) = 1 + 2 \cdot n_b(w)\} \ .$$

Construct a simple always terminating Turing machine M with $L(M) = L_6$. Give the complete state diagram. Indicate in which states the computation can terminate when the input does not belong to L_6, why the machine always terminates, and why it accepts the language L_6.

Exercise 7 (12 %). Let L be a language over alphabet Σ, and let x and y be strings over Σ.
(a) Assume L is decidable. Indicate what this means (give the definition). Prove that $L' = \{w \in \Sigma^* \mid xw \in L \land wy \notin L\}$ is decidable.
(b) Assume L is semi-decidable. Indicate what this means (give the definition). Prove that $L'' = \{w \in \Sigma^* \mid xw \in L \lor wy \in L\}$ is semi-decidable.

Exercise 8 (10 %). The Lecture Notes describe how to encode a Turing machine $M \in \mathit{TM0}$ by means of a string $R(M)$, and they describe a universal Turing machine that can simulate any Turing machine M thus encoded.
(a) Describe the class *TM0* of the machines that can be encoded in this way, and describe the encoding $R(M)$ for an arbitrary machine $M \in \mathit{TM0}$.
(b) Describe the language L_U accepted by this universal Turing machine in words, and in set notation.
(c) Is the language L_U decidable? Is it semi-decidable? Justify your answers.

Exercise 9 (12 %) Consider the language

$$L_9 = \{R(M) \mid M \in \mathit{TM0} \ \land \ 1001 \in L(M)\}.$$

(a) Prove that the language L_9 is not decidable.
(b) Is the language L_9 semi-decidable? Justify your answer.