Introduction to the Theory of Computation

Final exam

19 August 2016

Closed-book. Duration: 3h30.
Please answer each question on a separate sheet with your name and section. ***Motivate all your answers and give sufficient details.***

1. a) Is an infinite subset of a set that is non-denumerable necessarily non-denumerable?

   b) Is the union of two denumerable sets necessarily denumerable?

   c) Show, using a cardinality argument, that there must exist uncomputable functions from $\mathbb{N}$ to $\mathbb{N}$.

2. a) Give a DFA that accepts the language

   $$L_1 = \{w \mid w \in \{a,b\}^*,\ N_a(w) = 2 \mod 4\}$$

   where $N_\sigma(w)$ is the number of letters σ contained in the word w.

   b) Give a DFA that accepts the language

   $$L_2 = \{w \mid w \in \{a,b\}^*,\ aa \notin Fact(w)\}$$

   c) Give a regular grammar that generates $\overline{L_1 \cup L_2}$.

3. a) Using the pumping lemma, show that the language $\{a^{3n}b^nc^* \mid n \in \mathbb{N}\}$ is not regular.

   b) Let L_1 and L_2 be two regular languages over the same alphabet Σ. Is the language $L_1 \oplus L_2$ that contains the words that belong to only one of the two languages regular?

4. a) Is the language $L = \{a^ib^jc^k \mid i = j \text{ or } j = k \text{ where } i,j,k \geqslant 0\}$ context-free?

   b) Show that the intersection of two context-free languages is not necessarily context-free. Use this to deduce that the complement of a context-free language is not necessarily context-free. Give a sufficient criterion for the intersection of two context-free languages to be context-free.

5. a) For a Turing machine M, define the notions of *configuration*, *derivation*, *execution*, *accepted language* and *decided language*.

   b) Give a Turing machine that decides the language $L = \{a^n b^{2n} \mid n \geqslant 0\}$ defined over the alphabet $\Sigma = \{a, c, b\}$. Explain briefly the role of each state of the Turing machine.

6. a) Define *primitive recursive functions*.

   b) Show that the function SumSquares(n), that computes the sum of the squares from 0 up to n (e.g. SumSquares$(3) = 0^2+1^2+2^2+3^2$), is primitive recursive.

7. Let M be a Turing machine and q one of its states. Show that the problem that consist in determining whether there exists a word such that M stops in state q is undecidable.
   Hint: Use the existential halting-problem.

8. a) Consider $L \in NP$. Show that there exists a deterministic Turing machine M and a polynomial $p(n)$ such that M decides L and has a time complexity bounded by $2^{p(n)}$.

   b) State *Cook's theorem* and explain its importance.