# Introduction to the Theory of Computation

## Final exam

24 January 2017

*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 the set of all functions from $\{0,1\}$ to $\mathbb{N}$ denumerable?

   b) Is the set of all functions from $\mathbb{N}$ to $\{0,1\}$ denumerable?

2. Let $L$ be the language of the words $w$ over the alphabet $\{a,b,c\}$ that respect at least one of the following conditions:

   - $w$ contains one or more letters and begins and ends with the letter $b$
   - $N_a(w) + N_c(w)$ is a multiple of 3, where $N_\sigma(w)$ is the number of letters $\sigma$ in the word $w$.

   a) Give a NFA that accepts $L$.

   b) Give a DFA that accepts $L$.

   c) Give a regular grammar that generates $L$.

3. a) Let $L$ be the language over the alphabet $\{<,>,!\}$ generated by the following grammar where $S$ is the start symbol

   $$\begin{array}{l} S \to\ < S >\ \mid\ > S <\ \mid \varepsilon \\ E \to\ !E \mid \varepsilon; \end{array}$$

   i. Describe $L$.

   ii. Is $L$ a regular language?

   iii. Is $L$ a context-free language?

   b) Let $L_1$ be a regular language and $L_2$ a language such that $L_2 \subseteq L_1$. Is $L_2$ necessarily a regular language?

4. a) Give a pushdown-automaton that accepts the words over the alphabet $\{a, b, c\}$ where the number of a's is equal to the number of b's and where each letter a is followed by cc.

   b) Given a context-free grammar $G$, give an algorithm for checking if $L(G) = \varnothing$.

5. a) Give the definition of a non-deterministic Turing Machine and prove that any language that is accepted by a non-deterministic Turing Machine is also accepted by a deterministic Turing Machine.

   b) Why are Turing Machines of interest in computability theory even though they are quite different from the machines in current use?

6. a) Define the notion of *primitive recursive* functions as well as the concepts used in this definition.

   b) Show that $\text{AbsDiff}(a, b) = |a - b|$ is primitive recursive. Is AbsDiff $\mu$-recursive?

7. a) Define the classes R and RE. What can you say about a language $L$ and its complement $\overline{L}$ with respect to membership in R and RE?

   b) Let $M$ be a Turing Machine and $x$, $y$ and $z$ three words. Show that determining wheter all the words in the language $xy^*z$ are accepted by $M$ is undecidable.

8. a) Define the complexity class $NP$ and the complexity measures used in this definition.

   b) Let $L \in NP$. Is $L$ decidable? Prove your statement.