Free University of Bozen-Bolzano – Faculty of Computer Science
Master of Science in Computer Science
Theory of Computing – A.Y. 2006/2007
Final exam – 1/2/2007 – Part 2

Time: 90 minutes

This is a closed book exam: the only resources allowed are blank paper, pens, and your head. Explain your reasoning. Write clearly, in the sense of logic, language and legibility. The clarity of your explanations affects your grade. Write your name and ID on every solution sheet. Good luck!

Problem 2.1 [6 points] Decide, if possible, which of the following statements is TRUE and which is FALSE. You must give a brief explanation of your answer to receive full credit.

(a) For all languages L_1 and L_2, if L_1 is in P and L_2 is in NP, then $L_1 \cap L_2$ is in P.

(b) For all languages L_1 and L_2, if L_2 is in NP and $L_1 <_{poly} L_2$, then L_1 is in P.

(c) The class NP is closed under union.

(d) There exists a language L such that both L and $\overline{L}$ are recursively enumerable, but neither L nor $\overline{L}$ are recursive.

Problem 2.2 [6 points]

(a) Describe an algorithm to convert a context free grammar into Chomsky Normal Form.

(b) Illustrate the algorithm on the grammar $G = (\{S, A, B, C\}, \{a, b\}, P, S)$, where P consists of the following productions:

$$\begin{array}{lcl} S & \longrightarrow & A \mid ABa \mid AbA \\ A & \longrightarrow & Aa \mid \varepsilon \\ B & \longrightarrow & Bb \mid BC \\ C & \longrightarrow & CB \mid CA \mid bB \end{array}$$

Problem 2.3 [6 points] Let $L_e \subseteq \{0,1\}^*$ be the language of binary strings that have the same number of 0's and 1's. Construct a Turing Machine M_e that decides L_e, i.e., such that M_e always halts and $\mathcal{L}(M_e) = L_e$. Show the sequence of IDs of M_e on the accepted input string 1010 and on the non-accepted input string 1011.

Problem 2.4 [6 points] Let $M = (Q, \Sigma, \Gamma, \delta, q_0, \mathtt{B}, F)$ be a standard Turing Machine that accepts a language L, i.e., $\mathcal{L}(M) = L$. Informally, but precisely describe how to construct, from M a new Turing Machine M' that accepts a string $w \in \Sigma^*$ if and only if there is a substring of w in L. [*Hint*: Make use of standard TM constructions and extensions of the basic TM model, e.g., with non-determinism.]

Problem 2.5 [6 points] For a Turing Machine M with input alphabet $\Sigma = \{a, b\}$, let $\mathcal{E}(M)$ denote the encoding of M, and $\langle \mathcal{E}(M), w \rangle$ denote the encoding of M together with an input word w. Consider the language $L = \{\langle \mathcal{E}(M), w \rangle \mid M$, when started on an input word w, eventually prints the symbol a on two consecutive transitions $\}$.

(a) Show that L is recursively enumerable. [*Hint*: Make use of a universal TM.]

(b) Show that L is not recursive. [*Hint*: Exploit a reduction from the halting problem.]