Free University of Bozen-Bolzano – Faculty of Computer Science
Master of Science in Computer Science
Theory of Computing – A.Y. 2006/2007
Final exam – 12/6/2007 – 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) Assume that P $\neq$ NP. Then, for all languages L_1 and L_2, if L_1 is in P and L_2 is in NP but not in P, then $L_1 \cup L_2$ is in NP but not in P.

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

(c) The class NP is closed under complement.

(d) There exist two languages L_1 and L_2 such that both L_1 and L_2 are recursive, but $L_1 \cdot L_2$ is non-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:

$$\begin{array}{rcl} S & \longrightarrow & ABCa \mid aAbb \mid \varepsilon \\ A & \longrightarrow & \varepsilon \\ B & \longrightarrow & bB \mid b \mid AC \\ C & \longrightarrow & aCa \mid \varepsilon \end{array}$$

Problem 2.3 [6 points] Let $L_p \subseteq \{0, 1\}^*$ be the language of binary words that have the form ww_m, where w_m is obtained from w by inverting the order of symbols and by replacing each 0 with a 1, and each 1 with a 0. For example, if $w = 000101$, then $w_m = 010111$. Construct a TM M_p that decides L_p, i.e., such that M_p always halts and $\mathcal{L}(M_p) = L_p$. Show the sequence of IDs of M_p on the accepted input string 1010 and on the non-accepted input string 1001.

Problem 2.4 [6 points] Show that the class of recursively enumerable languages is closed under intersection. To do so, informally but precisely describe how to construct, from two Turing Machines M_1 and M_2, a new TM that accepts $\mathcal{L}(M_1) \cap \mathcal{L}(M_2)$. [*Hint*: Make use of standard TM constructions and extensions of the basic TM model, e.g., with multiple tapes or with non-determinism.]

Problem 2.5 [6 points] Consider a variant of the standard Turing Machine, called TM_{rw}, in which a machine M has two heads, a read-only head and a write-only head, that can move independently on the tape. At the beginning of the computation, the input word is written on the tape, surrounded by blanks, and both heads are placed on the left-most symbol of the input word. The transitions are specified through a transition function

$$\delta : Q \times \Gamma \longrightarrow Q \times \Gamma \times \{\mathit{left}, \mathit{right}\} \times \{\mathit{left}, \mathit{right}\}$$

where $\delta(p, x) = (q, y, d_r, d_w)$ means that, if M is in state p and the *read-only head* is over tape symbol x, then M changes state to q, the tape symbol under the *write-only head* is replaced by y, the read-only head moves by one cell in direction d_r, and the write-only head moves by one cell in direction d_w. Acceptance is defined as for standard TMs.

Show that TM_{rw} machines accept the class of recursively enumerable languages.

[*Hint*: For one direction, describe how the computation of a standard TM can be directly simulated by a TM_{rw}. For the other direction, exploit ideas similar to the simulation of multiple head/tape TMs by standard TMs.]