Albert-Ludwigs-Universität, Inst. für Informatik
Prof. Dr. Fabian Kuhn
P. Bamberger, Y. Maus,

Theoretical Computer Science - Bridging Course
Summer Term 2017
Exercise Sheet 7

Hand in (electronically or hard copy) by 12:15 pm, December 11th, 2017

Exercise 1: Decidability *(7 Points)*

Consider the following language

$$A_{\text{all}} = \{\langle M \rangle \mid \text{TM } M \text{ accepts all inputs}\}$$

Show that A_{all} is not decidable.
Hint: Assume that a TM M_c decides A_{all}. Then run the Turing machine M_c on some very wisely chosen input to decide the halting problem.

Sample Solution

Assume that the language is decidable, i.e., there is a TM M_c that decides the language A_{all}. We use the TM to create a Turing machine for the halting problem. This will be a contradiction!
For an input x we define a Turing machine M_x^* as follows: If $x = \langle M, w \rangle$ for some TM M and some words w then M_x^* ignores its input, simulates M on w and accepts afterwards, that is, the new Turing machine halts and accepts if and only if M halts on w. If x does not encode a TM and an input M^* simply ignores its input and rejects.
Now we construct a Turing machine M_{halting} which uses M_c as a subprogramm and decides the halting problem. On input x it construct the Turing machine M_x^*. Then it uses M_c on input $\langle M_x^* \rangle$ and accepts if M_c accepts, otherwise it rejects. We prove that M_{halting} decides the halting problem. If the input x for M_{halting} does not encode a TM and an input, M_x^* rejects, thus M_x^* is not accepted by M_c and M_{halting} rejects x.
If the input x for M_{halting} is of the format $\langle M, w \rangle$ for some TM M and a word w we have the following equivalences.

$$\begin{aligned}
&M_{\text{halting}} \text{ accepts } \langle M, w \rangle \\
\Leftrightarrow &M_c \text{ accepts } \langle M_{M,w}^* \rangle \\
\Leftrightarrow &M_{M,w}^* \text{ accepts every input} \\
\Leftrightarrow &M \text{ halts on } w
\end{aligned}$$

Thus M_{halting} solves the halting problem, a contradiction.

Exercise 2: Landau Notation *(2+2+3 Points)*

Prove or disprove the following statements

(a) $100\sqrt{n} \in O(0.001 \cdot n)$.

(b) $\log_2 3^n \in O(n)$.

(c) $2 \cdot n \in O(10 \cdot \sqrt{n})$.

Remark: There are thousands of similar exercises on the 'net'. Go through some of them to practice for the exam.

Sample Solution

(a) The claim is true. Let $c = 100000$ and $n_0 = 1$. Then we have for all $n \geq n_0$

$$100\sqrt{n} \leq c \cdot 0.001 \cdot \sqrt{n} \leq c \cdot 0.001 \cdot n.$$

(b) The claim is true. Let $c = \log_2 3 \approx 1.58$ and $n_0 = 1$. Then we have for all $n \geq n_0$ $\log_2 3^n = n \cdot log_2 3$

(c) The claim is false. Assume there is a $c > 0$ and n_0 such that $2 \cdot n \leq c \cdot 10\sqrt{n}$ for all $n \geq n_0$. Then we have for all $n \geq n_0$ that

$$2 \cdot n \leq c \cdot 10\sqrt{n} \Leftrightarrow n \leq c \cdot 5 \cdot \sqrt{n} \Leftrightarrow \sqrt{n} \leq 5c \Leftrightarrow n \leq 25c^2 \tag{1}$$

This is a contradiction for all $n \geq 25c^2$.

Exercise 3: Sort Functions by Asymptotic Growth *(6 Points)*

Sort the following functions by asymptotic growth using the $\mathcal{O}$-notation. Write $g <_{\mathcal{O}} f$ if $g \in \mathcal{O}(f)$ *and* $f \notin \mathcal{O}(g)$. Write $g =_{\mathcal{O}} f$ if $f \in \mathcal{O}(g)$ *and* $g \in \mathcal{O}(f)$.

n^2	$\sqrt{n}$	2^n	$\log(n^2)$
3^n	n^{100}	$\log(\sqrt{n})$	$(\log n)^2$
$\log n$	$10^{100}n$	$n!$	$n \log n$
$n \cdot 2^n$	n^n	$\sqrt{\log n}$	n

Sample Solution

	$\sqrt{\log n}$	$<_{\mathcal{O}}$	$\log(\sqrt{n})$	$=_{\mathcal{O}}$	$\log n$	$=_{\mathcal{O}}$	$\log(n^2)$
$<_{\mathcal{O}}$	$(\log n)^2$	$<_{\mathcal{O}}$	$\sqrt{n}$	$<_{\mathcal{O}}$	n	$=_{\mathcal{O}}$	$10^{100}n$
$<_{\mathcal{O}}$	$n \log n$	$<_{\mathcal{O}}$	n^2	$<_{\mathcal{O}}$	n^{100}	$<_{\mathcal{O}}$	2^n
$<_{\mathcal{O}}$	$n \cdot 2^n$	$<_{\mathcal{O}}$	3^n	$<_{\mathcal{O}}$	$n!$	$<_{\mathcal{O}}$	n^n