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

Theoretical Computer Science - Bridging Course
Summer Term 2017
Exercise Sheet 5

Hand in (electronically or hard copy) by 12:15 pm, November 27th, 2017

Exercise 1: Turing machines *(1+1+4+2 Points)*

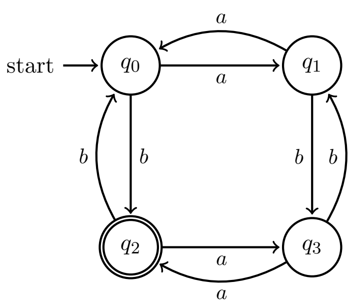
(a) Give a **comparison** of the set of languages recognized by **deterministic** Turing machines with the set of languages recognized by **non-deterministic** Turing machines.

(b) State **two differences** between **deterministic** and **non-deterministic** Turing machines.

(c) One can define a variant of the Turing machine which allows **three** actions of the read/write-head: $\{\boldsymbol{L}, \boldsymbol{R}, \boldsymbol{S}\}$, where $\boldsymbol{S}$ means that the head stands still during that step.

Let M_1 be a Turing machine **that uses head movements** $\{\boldsymbol{L}, \boldsymbol{R}, \boldsymbol{S}\}$. Give an **explicit** construction procedure that transfers M_1 into a Turing machine M_2 **that uses only head movements** $\{\boldsymbol{L}, \boldsymbol{R}\}$ and recognizes the same language, i.e. $L(M_1) = L(M_2)$.

(d) **Briefly** explain how to construct (or construct) a Turing machine for the language defined by the following deterministic finite automaton over the alphabet $\{a, b\}$.



Exercise 2: Constructing a Turing Machine *(6 Points)*

Let $L = \{\langle w\rangle, \langle w+1\rangle \mid w \in \mathbb{N}\}$, e.g., the word $\langle 6\rangle, \langle 7\rangle = 110, 111$ is contained in L. Design a Turing machine which accepts L. You do not need to provide a formal description of the Turing machine but your description has to be detailed enough to explain every possible step of a computation.
Remark: Here $\langle w\rangle$ is the binary encoding of the number w, e.g., the number 6 is going to be the string 110.

Exercise 3: Turing Machine *(flexible 1+1+1+1+1+1 Points)*

Let $M = (Q, \Sigma, \Gamma, \delta, q_0, q_{reject}, q_{accept})$ be a **deterministic** Turing machine over alphabet $\Sigma = \{0, 1\}$ with tape alphabet $\Gamma = \{0, 1, \sqcup\}$ that **always halts**. Furthermore, for all of the below exercises assume that M is **linearly space restricted**. You may assume that M never moves outside a range of $5n$ consecutive cells on the input tape when started on an input $w \in \Sigma^*$ of length n.

(a) Upper bound the **different read/write-head positions** that M could assume when running on an input of length n.

(b) Upper bound the **combinations of read/write-head positions and possible states** that M could assume on each position. Use the number of states $m := |Q|$ of M.

(c) Upper bound the **possible number of strings** that M could write on the tape when running on an input of length n with the tape restriction mentioned above.

(d) Upper bound the **possible number of configurations** of M on an input of length n with the tape restriction mentioned above.

(e) Explain why M can **never encounter a configuration twice**, when started on an input of size n. Keep in mind that M is a **deterministic decider**.

(f) Now, assume that the Turing machine M holds on every input. Upper bound its **maximum number of steps** on an input of length n.

***Remark**: Whenever we ask for upper bounds we want a bound as tight as possible. Giving looser upper bounds might yield partial points.*