Tik-79.148 **Spring 2001**
Introduction to Theoretical Computer Science
Tutorial 8
Solutions to Demonstration Exercises

Demonstration Exercises:

4. A Turing machine consists of a finite control, an input tape and a single head which reads the symbols from the tape and is also used to change the symbols on the tape. The control has a set of states, and it works like an ordinary finite automata. Instead, the function of the input tape is very different from the function of the input tape of a finite automata. The head can be moved both to the left and to the right, and it can also write new symbols on the tape. The tape has a left end (marked with left end symbol $\triangleright$), but it extends indefinitely to the right. The input to a Turing machine is written on the left end of the input tape, and to the right of the input the tape contains blank symbols $\sqcup$[1].

   The operation of a Turing machine is following:

   (a) The machine starts from an initial configuration $(s, \triangleright \sqcup w\underline{\sqcup})$ where s is the initial state, w the input and the underlined symbol indicates the head position. The slots to the right of the input contains blank symbols.

   (b) At each computation step the machine reads the symbol in the head position. Transition to a new state is chosen according to which symbol was read and the current state. At the same time either a new symbol is written to the tape or the head is moved.

   (c) At the end of the computation the machine moves a particular halting state h. The result of the computation is the string on the tape. It is also possible that the machine never reaches h and never halts.

   A few formal determinations can be given for a Turing machine. There appears two slightly different, but in practice equivalent, definitions in the material of this course. Both definitions has their own advantages, and neither is better than the other.

   (a) The older edition of the book [2] defines a Turing machine M as follows:

   $$M = (K, \Sigma, \delta, s),$$

   where

   - K is a finite set of states, halting state h *not* contained.
   - Σ is a finite alphabet, containing the blank symbol $\#$, but not containing the symbols L and R moving the head.
   - $\delta : (K \times \Sigma) \rightarrow (K \cup \{h\} \times \Sigma \cup \{L, R\})$ is the transition function.
   - $s \in K$ is the initial state.

[1] Notice that $\sqcup \neq e$.
[2] This definition is also used in the lecture notes.

	Old edition	New edition
Definition	$M = (K, \Sigma, \delta, s)$	$M = (K, \Sigma, \delta, s, H)$
Halting states	$h \notin K$	$H = \{h, y, n\} \subseteq K$
Transition function	$K \times \Sigma \to (K \cup \{h\}) \times (\Sigma \cup \{L, R\})$	$(K - H) \times \Sigma \to (K \times \Sigma \cup \{\leftarrow, \rightarrow\})$
Blank symbol	$\#$	$\sqcup$
Moving the head	$\{L, R\}$	$\{\leftarrow, \rightarrow\}$
Initial configuration	$(s, \#w\underline{\#})$	$(s, \rhd\underline{\sqcup}w)$
Deciding language L	$(s, \#w\underline{\#}) \vdash^* \begin{cases} (h, \#Y\underline{\#}), w \in L \\ (h, \#N\underline{\#}), w \notin L \end{cases}$	$(s, \rhd\underline{\sqcup}w) \vdash^* \begin{cases} (y, \rhd v), w \in L \\ (n, \rhd v), w \notin L \end{cases}$
Function $f(x) = y$	$(s, \#x\underline{\#}) \vdash^* (h, \#y\underline{\#})$	$(s, \rhd\underline{\sqcup}x) \vdash^* (h, \rhd\underline{\sqcup}y)$

Taulukko 1: Comparison of the two different definitions of a Turing machine

(b) The newer edition of the book defines a Turing machine as follows:

$$M = (K, \Sigma, \delta, s, H),$$

where

- K is a finite set of states.
- Σ is a finite aphabet, containing the blank symbol $\sqcup$ and the left end symbol $\rhd$, but not containing the symbols $\leftarrow$ and $\rightarrow$ moving the head.
- $\delta : (K - H) \times \Sigma \to K \times (\Sigma \cup \{\leftarrow, \rightarrow\}$ is the transition function such that:
  i. For all $q \in K - H$, if $\delta(q, \rhd) = (p, b)$, then $b = \rightarrow$.
  ii. For all $q \in K - H$ ja $a \in \Sigma$, if $\delta(q, a) = (p, b)$, then $b \neq \rhd$.
- $s \in K$ is the initial state.
- $H \subseteq K$ is the set of halting states.

In other words, the requirements for the transition function mean simply that always reading $\rhd$ the machine must move the head to the right and that the machine must not write the left end symbol on the tape.

The newer notation is used in these exercises but the starting and ending configurations are presented according to the older definition. Motivation to this convention is that combining the Turing machines is easier when the computation is determined as given in the older definition.

In the exercise it was given a Turing machine $M = (K, \Sigma, \delta, s, \{h\})$ for which

$$K = \{q_0, q_1, q_2, h\}$$
$$\Sigma = \{a, \sqcup, \rhd\}$$
$$s = q_0$$

and

q	σ	$\delta(q,\sigma)$	q	σ	$\delta(q,\sigma)$	q	σ	$\delta(q,\sigma)$
q_0	a	$(q_1, \leftarrow)$	q_1	a	$(q_2, \sqcup)$	q_2	a	(q_2, a)
q_0	$\sqcup$	$(q_0, \sqcup)$	q_1	$\sqcup$	$(h, \sqcup)$	q_2	$\sqcup$	$(q_0, \leftarrow)$
q_0	$\triangleright$	$(q_0, \rightarrow)$	q_1	$\triangleright$	$(q_1, \rightarrow)$	q_2	$\triangleright$	$(q_2, \rightarrow)$

The computation of the machine with different values of n is examined when the machine starts from the configuration $(q_0, \triangleright \sqcup a^n \underline{a})$:

n = 0:
$(q_0, \triangleright \sqcup \underline{a}) \vdash_M (q_1, \triangleright\underline{\sqcup}a) \vdash_M (h, \triangleright\underline{\sqcup}a)$

n = 1:
$(q_0, \triangleright \sqcup a\underline{a}) \vdash (q_1, \triangleright \sqcup \underline{a}a) \vdash (q_2, \triangleright \sqcup \underline{\sqcup}a) \vdash_M (q_0, \triangleright\underline{\sqcup} \sqcup a) \vdash^*_M (q_0, \triangleright\underline{\sqcup} \sqcup a)$

n = 2:
$(q_0, \triangleright \sqcup aa\underline{a}) \vdash_M (q_1, \triangleright \sqcup a\underline{a}a) \vdash_M (q_2, \triangleright \sqcup a\underline{\sqcup}a) \vdash_M (q_0, \triangleright \sqcup \underline{a} \sqcup a) \vdash_M (q_1, \triangleright\underline{\sqcup}a \sqcup a) \vdash_M (h, \triangleright\underline{\sqcup}a \sqcup a)$

n = 3:
$(q_0, \triangleright \sqcup aaa\underline{a}) \vdash_M (q_1, \triangleright \sqcup aa\underline{a}a) \vdash_M (q_2, \triangleright \sqcup aa\underline{\sqcup}a) \vdash_M (q_0, \triangleright \sqcup a\underline{a} \sqcup a) \vdash_M (q_1, \triangleright \sqcup \underline{a}a \sqcup a) \vdash_M (q_2, \triangleright \sqcup \underline{\sqcup}a \sqcup a) \vdash_M (q_0, \triangleright\underline{\sqcup} \sqcup a \sqcup a) \vdash^*_M (q_0, \triangleright\underline{\sqcup} \sqcup a \sqcup a)$

...

We notice that the operation of the machine depends on whether n is odd or even. In both cases the machine reads the tape from the right to the left and replaces alternate a with a blank symbol. When n is even the machine halts when reached the left end of the tape. With odd n values the machine never halts but finally remains in an endless loop of writing symbol $\sqcup$ at the beginning of the tape.

Even n: $(q_0, \triangleright \sqcup a^n \underline{a}) \vdash^*_M (h, \triangleright\underline{\sqcup}a(\sqcup a)^{n/2})$

Odd n: $(q_0, \triangleright \sqcup a^n \underline{a}) \vdash^*_M (q_0, \triangleright\underline{\sqcup}(\sqcup a)^{(n+1)/2})$

5. Turing machine M decides the language L if

$$(s, \triangleright \sqcup w\underline{\sqcup}) \vdash^*_M (h, \triangleright \sqcup Y\underline{\sqcup}) \text{ if } w \in L, \text{ ja}$$
$$(s, \triangleright \sqcup w\underline{\sqcup}) \vdash^*_M (h, \triangleright \sqcup N\underline{\sqcup}) \text{ if } w \notin L.$$

In other words, when it is tested whether a word w is a member of the language L the word w is given as input to the Turing machine deciding the language. If the word is a member of the language the machine eventually halts and the answer Y is on the tape. If the word is not a member of the language the answer N is written on the tape.

The language $\{w \in \{a,b\}^* \mid w\colon \text{ includes at least one } a\}$ is decided by the Turing machine

$$M = (K, \Sigma, \delta, s, \{h\})$$
$$K = \{q_0, q_1, q_2, q_3, q_4, q_5, h\}$$
$$\Sigma = \{a, b, Y, N, \sqcup, \triangleright\}$$
$$s = q_0$$

The transition function includes only the situations which are possible when the input has the right shape:

q	σ	$\delta(q,\sigma)$
q_0	$\sqcup$	$(q_1, \leftarrow)$
q_1	a	$(q_2, \sqcup)$
q_1	b	$(q_0, \sqcup)$
q_1	$\sqcup$	$(q_5, \rightarrow)$
q_2	$\sqcup$	$(q_3, \leftarrow)$
q_3	a	$(q_2, \sqcup)$
q_3	b	$(q_2, \sqcup)$
q_3	$\sqcup$	$(q_4, \rightarrow)$
q_4	$\sqcup$	(q_4, Y)
q_4	Y	$(h, \rightarrow)$
q_5	$\sqcup$	(q_5, N)
q_5	N	$(h, \rightarrow)$

The machine reads the word from the right to the left. As long as it no a is read the machine changes between states q_0 and q_1 and concurrently empties the tape. If the word ends before the first a is found the machine moves to state q_5 and writes N on the tape. If a is read the machine empties rest of the tape by using the states q_2 and q_3, moves to state q_4 and writes Y on the table.

For instance:
$(q_0, \triangleright \sqcup ab\underline{\sqcup}) \vdash_M (q_1, \triangleright \sqcup a\underline{b}) \vdash_M (q_0, \triangleright \sqcup a\underline{\sqcup}) \vdash_M (q_1, \triangleright \sqcup \underline{a}) \vdash_M$
$(q_2, \triangleright \sqcup \underline{\sqcup}) \vdash_M (q_3, \triangleright\underline{\sqcup}) \vdash_M (q_4, \triangleright \sqcup \underline{\sqcup}) \vdash_M (q_4, \triangleright \sqcup \underline{Y}) \vdash_M (h, \triangleright \sqcup Y\underline{\sqcup})$