

CS 341 Homework 17 Turing Machines

1. Let $M = (K, \Sigma, \delta, s, \{h\})$, where

$$K = \{q_0, q_1, h\},$$

$$\Sigma = \{a, b, \square, \diamond\},$$

$$s = q_0,$$

and δ is given by the following table,

q	σ	$\delta(q, \sigma)$
q_0	a	(q_1, b)
q_0	b	(q_1, a)
q_0	\square	(h, \square)
q_0	\diamond	(q_0, \rightarrow)
q_1	a	(q_0, \rightarrow)
q_1	b	(q_0, \rightarrow)
q_1	\square	(q_0, \rightarrow)
q_1	\diamond	(q_1, \rightarrow)

(a) Trace the computation of M starting from the configuration $(q_0, \diamond \underline{a} b b b a)$.

(b) Describe informally what M does when started in q_0 on any square of a tape.

2. Repeat Problem 1 for the machine $M = (K, \Sigma, \delta, s, \{h\})$, where

$$K = \{q_0, q_1, q_2, h\},$$

$$\Sigma = \{a, b, \square, \diamond\},$$

$$s = q_0,$$

and δ is given by the following table (the transitions on \diamond are $\delta(q, \diamond) = (q, \diamond)$, and are omitted).

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

Start from the configuration $(q_0, \diamond \underline{a} b b \square b b \square \square \square \underline{a} b a)$.

3. Let M be the Turing machine $M = (K, \Sigma, \delta, s, \{h\})$, where

$$K = \{q_0, q_1, q_2, h\},$$

$$\Sigma = \{a, \square, \diamond\},$$

$$s = q_0,$$

and δ is given by the following table.

Let $n \geq 0$. Describe carefully what M does when started in the configuration $(q_0, \diamond \square a^n \underline{a})$.

q	σ	$\delta(q, \sigma)$
q ₀	a	(q ₁ , ←)
q ₀	□	(q ₀ , □)
q ₀	◇	(q ₀ , →)
q ₁	a	(q ₂ , □)
q ₁	□	(h, □)
q ₁	◇	(q ₁ , →)
q ₂	a	(q ₂ , a)
q ₂	□	(q ₀ , ←)
q ₂	◇	(q ₂ , →)

4. Design and write out in full a Turing machine that scans to the right until it finds two consecutive a's and then halts. The alphabet of the Turing machine should be {a, b, □, ◇}.

5. Give a Turing machine (in our abbreviated notation) that takes as input a string $w \in \{a, b\}^*$ and squeezes out the a's. Assume that the input configuration is $(s, \diamond \square w)$ and the output configuration is $(h, \diamond \square w')$, where $w' = w$ with all the a's removed.

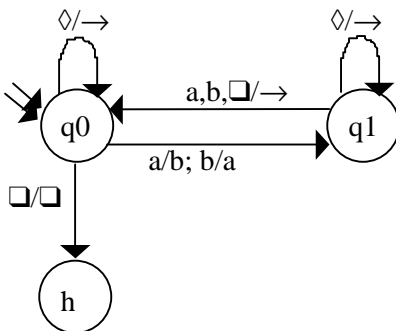
6. Give a Turing machine (in our abbreviated notation) that shifts its input two characters to the right.

Input: $\square w \square$
Output: $\square \square w \square$

7. (L & P 5.7.2) Show that if a language is recursively enumerable, then there is a Turing machine that enumerates it without ever repeating an element of the language.

Solutions

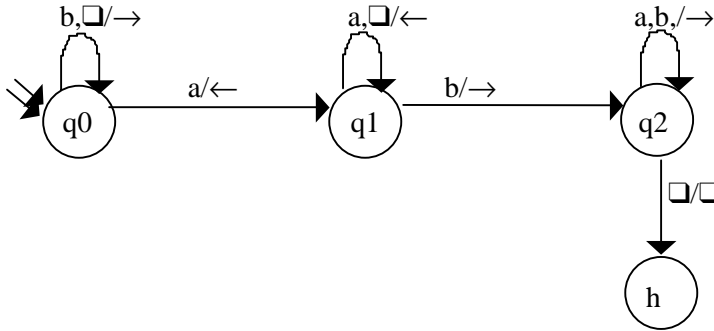
1. (a)



q₀, ◇aabbba
q₁, ◇babbba
q₀, ◇babbba
q₁, ◇bbbbbba
q₀, ◇bbbbbbba
q₁, ◇bbabbba
q₀, ◇bbabba
q₁, ◇bbaaba
q₀, ◇bbaaba
q₁, ◇bbaaaa
q₀, ◇bbaaaaa
q₁, ◇bbaaaab
q₀, ◇bbaaab□
h, ◇bbaaab□

(b) Converts all a's to b's, and vice versa, starting with the current symbol and moving right.

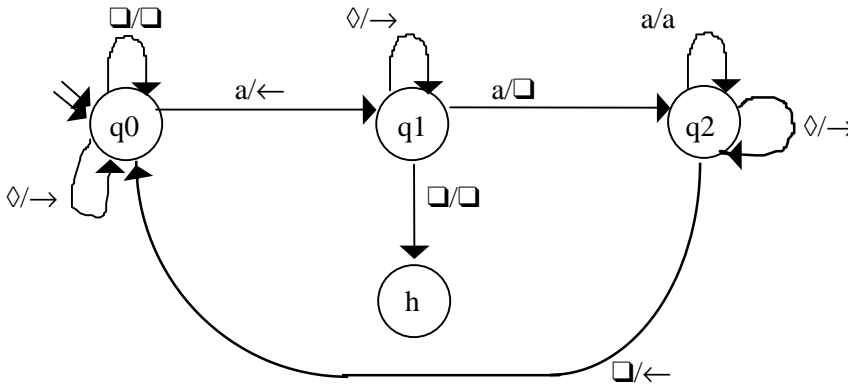
2. (a)



$q_0, \diamond \underline{a} b b \square b b \square \square \square a b a$
 $q_0, \diamond \underline{a} b b \square b b \square \square \square a b a$
 $q_0, \diamond \underline{a} b b \square b b \square \square \square a b a \dots$
 $q_1, \diamond \underline{a} b b \square b b \square \square \square a b a$
 $q_1, \diamond \underline{a} b b \square b b \square \square \square a b a \dots$
 $q_2, \diamond \underline{a} b b \square b b \square \square \square a b a$
 $h, \diamond \underline{a} b b \square b b \square \square \square a b a$

(b) M goes right until it finds an a, then left until it finds a b, then right until it finds a blank.

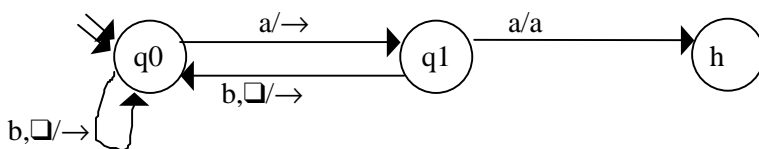
3.



$q_0, \diamond \square a a a a a$
 $q_1, \diamond \square a a a a a$
 $q_2, \diamond \square a a a a a$
 $q_0, \diamond \square a a a a a$
 $q_1, \diamond \square a a a a a$
 $q_2, \diamond \square a a a a a$
 $q_0, \diamond \square a a a a a$
 $q_1, \diamond \square a a a a a$
 $h, \diamond \square a a a a a$

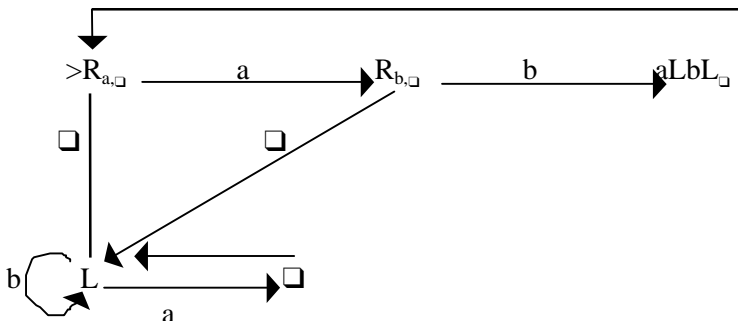
M changes every other a, moving left from the start, to a blank. If n is odd, it loops. If n is even, it halts.

4.

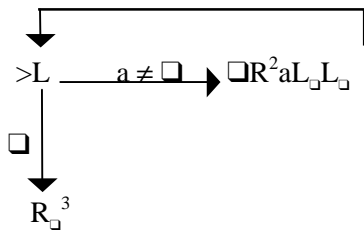


$M = (K, \Sigma, \delta, s, \{h\})$, where
 $K = \{q_0, q_1, h\}$,
 $\Sigma = \{a, b, \square, \diamond\}$,
 $s = q_0$

5. The idea here is that first we'll push all b's to the left, thus squeezing all the a's to the right. Then we'll just replace the a's with blanks. In more detail: scan the string from left to right. Every time we find an a, if there are any b's to the right of it we need to shift them left. So scan right, skipping as many a's as there are. When we find a b, swap it with the last a. That squeezes one a further to the right. Go back to the left end and repeat. Eventually all the a's will come after all the b's. At that point, when we look for a b following an a, all we'll find is a blank. At that point, we just clean up by rewriting all the a's as blanks.



6. The idea is to start with the rightmost character of w , rewrite it as a blank, then move two squares to the right and plunk that character back down. Then scan left for the next leftmost character, do the same thing, and so forth.



7. Suppose that M is the Turing machine that enumerates L . Then construct M^* to enumerate L with no repetitions: M^* will begin by simulating M . But whenever M outputs a string, M^* will first check to see if the string has been output before (see below). If it has, it will just continue looking for strings. If not, it will output it, and it will also write the string, with $\#$ in front of it, at the right end of the tape. To check whether a string has been output before, M^* just scans its tape checking for the string it is about to output.