23/10/2009
E 2.1

Exercise (Example 8.2 from textbook)

Construct a Turing Machine accepting the language

$$\{0^n 1^n \mid n \geq 1\}$$

solution

The idea is that the TM M that we construct reads the leftmost 0, turns it into X, and moves right until it reaches a 1, that is turned into Y. Then the head moves left again to the leftmost 0 (on the right to a X), and starts again until all 0's and 1's are turned into X's and Y's respectively.
If the input is not in $0^* 1^*$, M will fail to find a move and it won't accept. If M changes the last 0 and the last 1 in the same round, it will go into the final state and accept.

$Q = \{q_0, q_1, q_2, q_3, q_4\}$

$\Sigma = \{0, 1\}$

$\Gamma = \{0, 1, X, Y, b\}$ (b denotes blank symbol)

q_0 : start state

$F = \{q_4\}$

In q_0 is the state in which M is when the head preceds the leftmost 0. In state q_1, M moves right skipping 0's and Y's until it gets to a 1. In state q_2, M moves left while skipping Y's and 0's again, until it gets to a X and goes again in q_0.

E 2.2

Starting from q_0, if a Y is read instead of a 0, M goes in q_3 and moves right: if a 1 is found, then there are more 1's than 0's; if a $\not{b}$ is read, then the initial string is accepted (transition to q_4).

	0	1	X	Y	$\not{b}$
q_0	(q_1, X, R)	—	—	(q_3, Y, R)	—
q_1	$(q_1, 0, R)$	(q_2, Y, L)	—	(q_1, Y, R)	—
q_2	$(q_2, 0, L)$	—	(q_0, X, R)	(q_2, Y, L)	—
q_3	—	—	—	(q_3, Y, R)	$(q_4, \not{b}, R)$
q_4	—	—	—	—	—

Exercise

Show the computation of the TM above when the input string is:

(a) 00

(b) 000111

solution

(a) $q_0 00 \vdash X q_1 0 \vdash X 0 q_1$

and the TM halts

(b) $q_0 000111 \vdash X q_1 00111 \vdash X 0 q_1 0111 \vdash$
$X 0 0 q_1 111 \vdash X 0 q_2 0 Y 11 \vdash X q_2 0 0 Y 11 \vdash q_2 X 0 0 Y 11 \vdash$
$X q_0 0 0 Y 11 \vdash X X q_1 0 Y 11 \vdash X X 0 q_1 Y 11 \vdash X X 0 Y q_1 11 \vdash$
$X X 0 q_2 Y Y 1 \vdash X X q_2 0 Y Y 1 \vdash X q_2 X 0 Y Y 1 \vdash X X q_0 0 Y Y 1 \vdash$
$X X X q_1 Y Y 1 \vdash X X X Y q_1 Y 1 \vdash X X X Y Y q_1 1 \vdash X X X Y q_2 Y Y \vdash$
$X X X q_2 Y Y Y \vdash X X q_2 X Y Y Y \vdash X X X q_0 Y Y Y \vdash X X X Y q_3 Y Y \vdash$
$X X X Y Y q_3 Y \vdash X X X Y Y Y q_3 \not{b} \vdash X X X Y Y Y \not{b} q_4 \not{b}$

Exercise (8.2.3 from textbook):

Design a Turing Machine that takes as input a number N in binary and turns it into $N+1$ (in binary); the number N is preceded by the symbol \$, which may be destroyed during the computation. For example, \$111 is turned into 1000; \$1001 is turned into \$1010.

solution

The idea is to toggle the rightmost digit, and, from right to left, all consecutive 1's until we get to the first 0 (which is also toggled). If there is no 0 to be toggled, a 1 is added on the left of the first digit (i.e., in place of the \$).

We need three states, where only q_2 is the final state; we briefly describe what the TM does in the different states.

q_0 : the TM goes right until it reaches b, after the rightmost digit. When b is reached, the TM goes into q_1.

q_1 : goes left toggling all 1's and the first 0 (from right); when 0 or \$ is reached, the symbol is turned into 1.

q_2 : final state; the TM does nothing

	\$	0	1	b
q_0	$(q_0, \$, R)$	$(q_0, 0, R)$	$(q_0, 1, R)$	(q_1, b, L)
q_1	$(q_2, 1, L)$	$(q_2, 1, L)$	$(q_1, 0, L)$	—
q_2	—	—	—	—

Exercise (8.22 from textbook):

Design Turing machines accepting the following languages:

$$\{ w \in \{0,1\}^* \mid w \text{ has an equal number of 0's and 1's} \}$$

Solution

The idea is that the head of our TM M moves back and forth on the tape, "deleting" one 0 for each 1; if there are no 0's and 1's in the end, the string is accepted.
When in state q_1, M has found a 1 and looks for a 0; in state q_2 is the other way around.
Note then the head never moves left of any X, so that there are never unmatched 0's and 1's on the left of an X.
From initial state q_0, M picks up a 0 or a 1 and turns it into X. The only final state is q_4. In state q_3 M moves head left looking for the rightmost X.

	0	1	b	X	Y
q_0	(q_2, X, R)	(q_1, X, R)	(q_4, b, R)	—	(q_0, Y, R)
q_1	(q_3, Y, L)	$(q_1, 1, R)$	—	—	(q_1, Y, R)
q_2	$(q_2, 0, R)$	(q_3, Y, L)	—	—	(q_2, Y, R)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	—	(q_0, X, R)	(q_3, Y, L)
q_4	—	—	—	—	—

Exercise (8.4.2 from textbook)

Consider the following NTM:

$$M = (\{q_0, q_1, q_2\}, \{0,1\}, \{0,1,\not b\}, \delta, q_0, \not b, \{q_2\})$$

with δ defined as follows

	0	1	$\not b$
q_0	$\{(q_0,1,R)\}$	$\{(q_1,0,R)\}$	$\emptyset$
q_1	$\{(q_1,0,R),(q_0,0,L)\}$	$\{(q_1,1,R),(q_0,1,L)\}$	$\{(q_2,\not b,R)\}$
q_2	$\emptyset$	$\emptyset$	$\emptyset$

Show the ID's reached by M when the input is (a) 01.
(b) 011

Solution

(a)

$q_0 01$
|
$1 q_0 1$
|
$1 0 q_1 \not b$
|
$1 0 \not b q_2 \not b$

Note that here we do not branch.

E 2.6

(b)

$$
\begin{tikzcd}[column sep=small]
 & q_0 011 \arrow[d, dash] & \\
 & 1 q_0 11 \arrow[d, dash] & \\
 & 10 q_1 1 \arrow[dl, dash] \arrow[dr, dash] & \\
101 q_1 \not{b} \arrow[d, dash] & & 1 q_0 01 \arrow[d, dash] \\
101\not{b} q_2 \not{b} & & 11 q_0 1 \arrow[d, dash] \\
 & & 110 q_1 \not{b} \arrow[d, dash] \\
 & & 110 \not{b} q_2 \not{b}
\end{tikzcd}
$$

E 2.7

Exercise (8.4.5 from textbook)

Suppose you have a tape with all ␢'s except a single \$, with the head in some (unknow) position (a) Write a NTM able to enter into a final state (starting from initial state) by scanning \$.

(b) Then, write a deterministic TM doing the same job.

solution

(a) The TM just needs to guess whether \$ is on the left or on the right. We call q, q_f the two states (q_f is final).

$$\delta(q, ␢) = \{(q, ␢, L), (q, ␢, R)\}$$

$$\delta(q, \$) = \{(q_f, \$, R)\}$$

(b) The deterministic TM goes back and forth examining one new position on the tape on the left, and then one on the right; marked symbols are turned from ␢ to #.

	␢	#	\$
q_0	$(q_1, \#, L)$	$(q_0, \#, R)$	$(q_2, \$, R)$
q_1	$(q_0, \#, R)$	$(q_1, \#, L)$	$(q_2, \$, R)$
q_2	—	—	—

In q_0, the TM looks for the next ␢ on the right, while in q_1 it looks for the next one on the left. When a ␢ is reached, it is turned into # and the search starts over in the opposite direction.