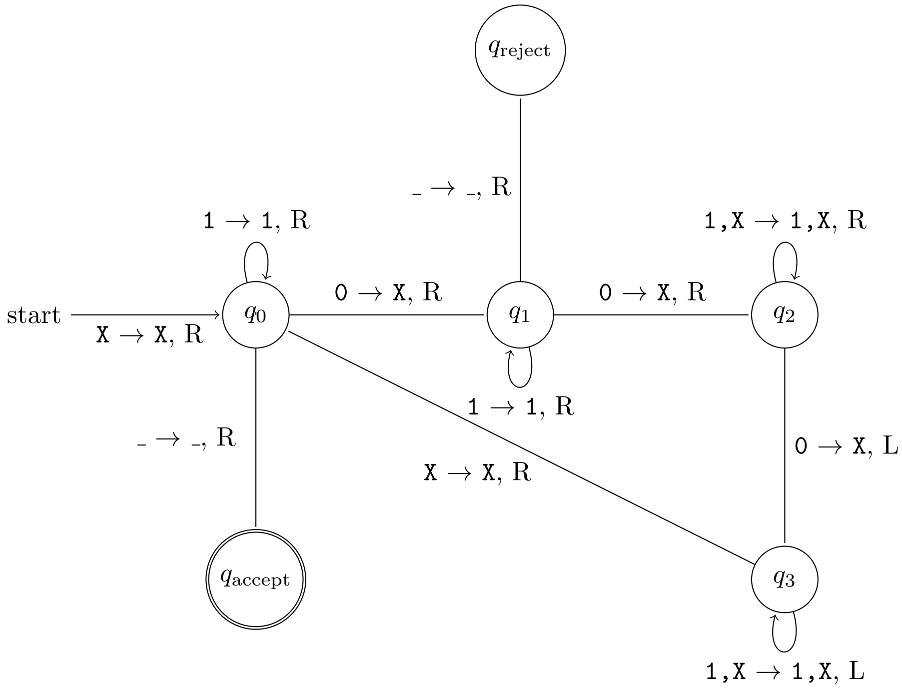
Turing Machine

Jan C. Bierowiec

The language $\{w \in \{0,1\}^* \mid \text{the number of 0s in } w \text{ is even}\}$.

1 State Diagram

Below is the state diagram of the TM that accepts strings with an even number of 0s:



2 Formal Description

The TM can be described as a 7-tuple:

$$M = (Q, \Sigma, \Gamma, \delta, q_0, \sqcup, F)$$

where:

$Q = \{q_0, q_1, q_2, q_3, q_{\text{accept}}, q_{\text{reject}}\}$	the set of states
$\Sigma = \{0, 1\}$	the input alphabet
$\Gamma = \{0, 1, X, \sqcup\}$	the tape alphabet
q_0	the start state
$\sqcup$	blank symbol
$F = \{q_{\text{accept}}\}$	the set of accepting states
δ	the transition function:

$\delta(q, a)$	(Write, Move, Next State)
$\delta(q_0, 0)$	$(X,\ R,\ q_1)$
$\delta(q_0, 1)$	$(1,\ R,\ q_0)$
$\delta(q_0, X)$	$(X,\ R,\ q_0)$
$\delta(q_0, \sqcup)$	$(\sqcup,\ R,\ q_{\text{accept}})$
$\delta(q_1, 0)$	$(X,\ R,\ q_2)$
$\delta(q_1, 1)$	$(1,\ R,\ q_1)$
$\delta(q_1, X)$	$(X,\ R,\ q_1)$
$\delta(q_1, \sqcup)$	$(\sqcup,\ R,\ q_{\text{reject}})$
$\delta(q_2, 0)$	$(0,\ L,\ q_3)$
$\delta(q_2, 1)$	$(1,\ R,\ q_2)$
$\delta(q_2, X)$	$(X,\ R,\ q_2)$
$\delta(q_2, \sqcup)$	$(\sqcup,\ R,\ q_{\text{reject}})$
$\delta(q_3, 0)$	$(0,\ L,\ q_3)$
$\delta(q_3, 1)$	$(1,\ L,\ q_3)$
$\delta(q_3, X)$	$(X,\ L,\ q_3)$
$\delta(q_3, \sqcup)$	$(\sqcup,\ R,\ q_0)$

How the TM Works

The Turing Machine begins in the start state q_0 and scans the tape from left to right. Its goal is to match pairs of 0s by marking them with an X, while skipping over 1s and already-marked positions.

- In q_0, the machine searches for the first unmarked 0. Upon finding one, it replaces it with X and transitions to q_1, continuing to move right to search for a second 0 to complete the pair.
- In q_1, when a second 0 is found, it is also marked with X, and the machine transitions to q_2 to continue scanning right. If instead the machine reaches the end of the tape before finding a second 0, it transitions to q_{reject}, indicating an unmatched (odd) 0.
- In q_2, the machine keeps moving right past any symbols until it finds a blank symbol ($\sqcup$), then transitions to q_3 (notated here as the "rewind" state), moving left to return to the beginning of the tape.
- In q_3, the machine continues moving left until it reaches the beginning of the tape (denoted by $\sqcup$), at which point it transitions back to q_0 to search for the next pair of unmarked 0s.

The process repeats until no more unmarked 0s remain. If all 0s have been paired (i.e., marked as X) and the machine reads only 1s, Xs, or blanks, it halts in state q_{accept}.

The TM accepts a string if:

- it reaches state q_{accept},
- all 0s have been matched and marked as X,
- the number of 0s in the input is even.

Example Strings and Traces

Accepted Strings: ε, `1`, `11`, `0011`, `1010`
Rejected Strings: `0`, `10`, `011`, `000`

Trace for `1010`:

$$
\begin{aligned}
&\text{Step 0: } q_0,\ \underline{1}010 \to q_0,\ 1,\ R\\
&\text{Step 1: } q_0,\ 1\underline{0}10 \to q_1,\ X,\ R\\
&\text{Step 2: } q_1,\ 1X\underline{1}0 \to q_1,\ 1,\ R\\
&\text{Step 3: } q_1,\ 1X1\underline{0} \to q_2,\ X,\ R\\
&\text{Step 4: } q_2,\ 1X1X\underline{\sqcup} \to q_3,\ \sqcup,\ L\\
&\text{Step 5: } q_3,\ 1X1\underline{X} \to q_3,\ X,\ L\\
&\text{Step 6: } q_3,\ 1X\underline{1}X \to q_3,\ 1,\ L\\
&\text{Step 7: } q_3,\ 1\underline{X}1X \to q_3,\ X,\ L\\
&\text{Step 8: } q_3,\ \underline{1}X1X \to q_3,\ 1,\ L\\
&\text{Step 9: } q_3,\ \underline{\sqcup}1X1X \to q_0,\ \sqcup,\ R\\
&\text{Step 10: } q_0,\ \sqcup\underline{1}X1X \to q_0,\ 1,\ R\\
&\text{Step 11: } q_0,\ 1\underline{X}1X \to q_0,\ X,\ R\\
&\text{Step 12: } q_0,\ 1X\underline{1}X \to q_0,\ 1,\ R\\
&\text{Step 13: } q_0,\ 1X1\underline{X} \to q_0,\ X,\ R\\
&\text{Step 14: } q_0,\ 1X1X\underline{\sqcup} \to q_{\text{accept}},\ \sqcup,\ R
\end{aligned}
$$

Final tape: `_1X1X_`
Final state: q_{accept}
Accepted

3 Context-Free Grammar (CFG)

The CFG that generates the same language (even number of 0s) is defined as:

$$G = (V, \Sigma, R, S)$$

where:

$V = \{S, A\}$	the set of variables
$\Sigma = \{0, 1\}$	the set of terminal symbols
S	the start symbol
R	the set of production rules

Production rules:

$$
\begin{aligned}
S &\to 1S \mid 0A \mid \varepsilon\\
A &\to 1A \mid 0S
\end{aligned}
$$

This CFG generates all strings over $\{0,1\}^*$ with an even number of 0s.

4 Regular Expression

Since the language is regular, it can also be described by the following regular expression:

$$(1^*(01^*01^*)^*)^*$$

This expression generates all strings with pairs of 0s, possibly separated by 1s.