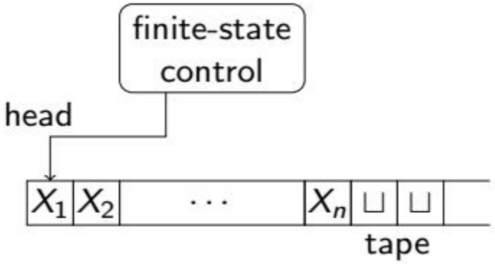
Turing Machines



- Unrestricted memory: an infinite tape
  - A finite state machine that reads/writes symbols on the tape
  - Can read/write anywhere on the tape
  - Tape is infinite in one direction only (other variants possible)
- Initially, tape has input and the machine is reading (i.e., tape head is on) the leftmost input symbol.
- Transition (based on current state and symbol under head):
  - Change control state
  - Overwrite a new symbol on the tape cell under the head
  - Move the head left, or right.

Example

```
0 1 1 0 0 0 # 0 1 1 0 0 0 ⊔ ...
x 1 1 0 0 0 # 0 1 1 0 0 0 ⊔ ...
x 1 1 0 0 0 # x 1 1 0 0 0 ⊔ ...
x 1 1 0 0 0 # x 1 1 0 0 0 ⊔ ...
x x 1 0 0 0 # x 1 1 0 0 0 ⊔ ...
x x x x x x # x x x x x x ⊔ ...
                          accept
```

- Let M_1 be a Turing machine that tests if an input string is in the language B, where $B = \{w\#w | w \in \{0,1\}^*\}$.
- M_1 zig-zags across the tape: if no $\#$ is found, *reject*. Cross off symbols as they are checked to keep track.
- When all symbols to the left of the $\#$ have been crossed off, check for any remaining symbols to the right of the $\#$. If any symbol remained, *reject*, otherwise *accept*.

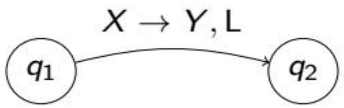
Turing Machines

Formal Definition

A Turing machine is $M = (Q, \Sigma, \Gamma, \delta, q_0, q_{\text{acc}}, q_{\text{rej}})$ where

- Q is a finite set of control states
- Σ is a finite set of input symbols
- $\Gamma \supseteq \Sigma$ is a finite set of tape symbols. Also, a blank symbol $\sqcup \in \Gamma \setminus \Sigma$
- $q_0 \in Q$ is the initial state
- $q_{\text{acc}} \in Q$ is the accept state
- $q_{\text{rej}} \in Q$ is the reject state, where $q_{\text{rej}} \neq q_{\text{acc}}$
- $\delta : Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$ is the transition function. Given the current state and symbol being read, the transition function describes the next state, symbol to be written and direction (left or right) in which to move the tape head.

Transition Function



$\delta(q_1, X) = (q_2, Y, L)$: Read transition as "the machine when in state q_1, and reading symbol X under the tape head, will move to state q_2, overwrite X with Y, and move its tape head to the left"

- In fact $\delta : (Q \setminus \{q_{\text{acc}}, q_{\text{rej}}\}) \times \Gamma \to Q \times \Gamma \times \{L, R\}$. No transition defined after reaching q_{acc} or q_{rej}
- Transitions are deterministic
- Convention: if $\delta(q, X)$ is not explicitly specified, it is taken as leading to q_{rej}, i.e., say $\delta(q, X) = (q_{\text{rej}}, \sqcup, R)$

Configurations

The configuration (or "instantaneous description") contains all the information to exactly capture the "current state of the computation"

$$X_1X_2\cdots X_{i-1}qX_i\cdots X_n$$

- Includes the current state: q
- Position of the tape head: Scanning i^{th} symbol X_i
- Contents of all the tape cells till the rightmost nonblank symbol. This is will always be finitely many cells. Those symbols are $X_1X_2\cdots X_n$, where $X_n \neq \sqcup$ unless the tape head is on it.

Single Step

Definition

We say one configuration (C_1) yields another (C_2), denoted as $\mathrm{C}_1 \vdash \mathrm{C}_2$, if one of the following holds.

- If $\delta(q, X_i) = (p, Y, \mathrm{L})$ then

  $$X_1X_2\cdots X_{i-1}qX_iX_{i+1}\cdots X_n \vdash X_1X_2\cdots X_{i-2}pX_{i-1}YX_{i+1}\cdots X_n$$

  Boundary Cases:
  - If $i = 1$ then $qX_1X_2\cdots X_n \vdash pYX_2\cdots X_n$
  - If $i = n$ and $Y = \sqcup$ then $X_1\cdots X_{n-1}qX_n \vdash X_1\cdots pX_{n-1}$
- If $\delta(q, X_i) = (p, Y, \mathrm{R})$ then

  $$X_1X_2\cdots X_{i-1}qX_iX_{i+1}\cdots X_n \vdash X_1X_2\cdots X_{i-1}YpX_{i+1}\cdots X_n$$

  Boundary Case:
  - If $i = n$ then $X_1\cdots X_{n-1}qX_n \vdash X_1\cdots X_{n-1}Yp\sqcup$

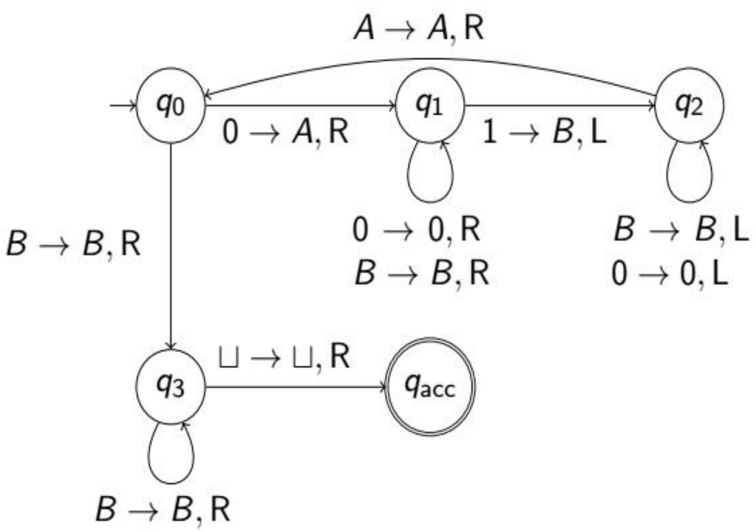
Example 1: TM for $\{0^n1^n \mid n > 0\}$

Design a TM to accept the language $L = \{0^n1^n \mid n > 0\}$

High level description

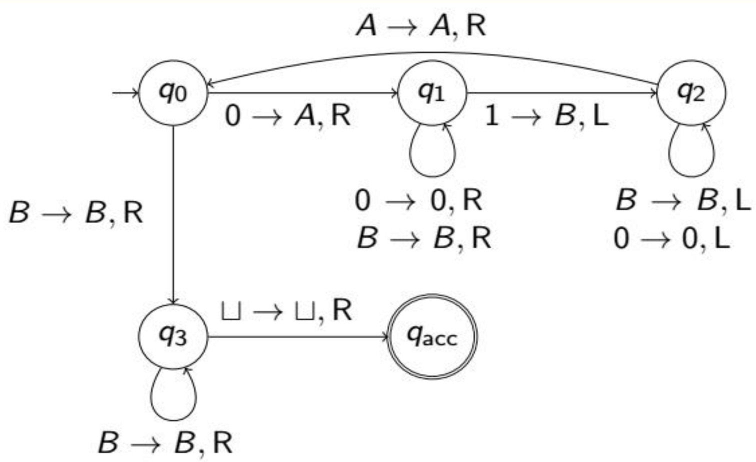
```
On input string w
    while there are unmarked 0s, do
        Mark the left most 0
        Scan right till the leftmost unmarked 1;
            if there is no such 1 then crash
        Mark the leftmost 1
    done
    Check to see that there are no unmarked 1s;
        if there are then crash
    accept
```

Example 1: TM for $\{0^n1^n \mid n > 0\}$



- Accepts input 0011: $q_00011 \vdash$

Example 1: TM for $\{0^n1^n \mid n > 0\}$



- Accepts input 0011: $q_00011 \vdash Aq_1011 \vdash A0q_111 \vdash Aq_20B1 \vdash q_2A0B1 \vdash Aq_00B1 \vdash AAq_1B1 \vdash AABq_11 \vdash AAq_2BB \vdash Aq_2ABB \vdash AAq_0BB \vdash AABq_3B \vdash AABBq_3\sqcup \vdash AABB\sqcup q_{\text{acc}}\sqcup$
- Rejects input 00: $q_000 \vdash Aq_10 \vdash A0q_1\sqcup \vdash A0\sqcup q_{\text{rej}}\sqcup$

Example: $\{0^n1^n \mid n > 0\}$

Formal Definition

The machine is $M = (Q, \Sigma, \Gamma, \delta, q_0, q_{acc}, q_{rej})$ where

- $Q = \{q_0, q_1, q_2, q_3, q_{acc}, q_{rej}\}$
- $\Sigma = \{0, 1\}$, and $\Gamma = \{0, 1, A, B, \sqcup\}$
- δ is given as follows

$$\begin{array}{ll}
\delta(q_0, 0) = (q_1, A, R) & \delta(q_0, B) = (q_3, B, R) \\
\delta(q_1, 0) = (q_1, 0, R) & \delta(q_1, B) = (q_1, B, R) \\
\delta(q_1, 1) = (q_2, B, L) & \delta(q_2, B) = (q_2, B, L) \\
\delta(q_2, 0) = (q_2, 0, L) & \delta(q_2, A) = (q_0, A, R) \\
\delta(q_3, B) = (q_3, B, R) & \delta(q_3, \sqcup) = (q_{acc}, \sqcup, R)
\end{array}$$

In all other cases, $\delta(q, X) = (q_{rej}, \sqcup, R)$.

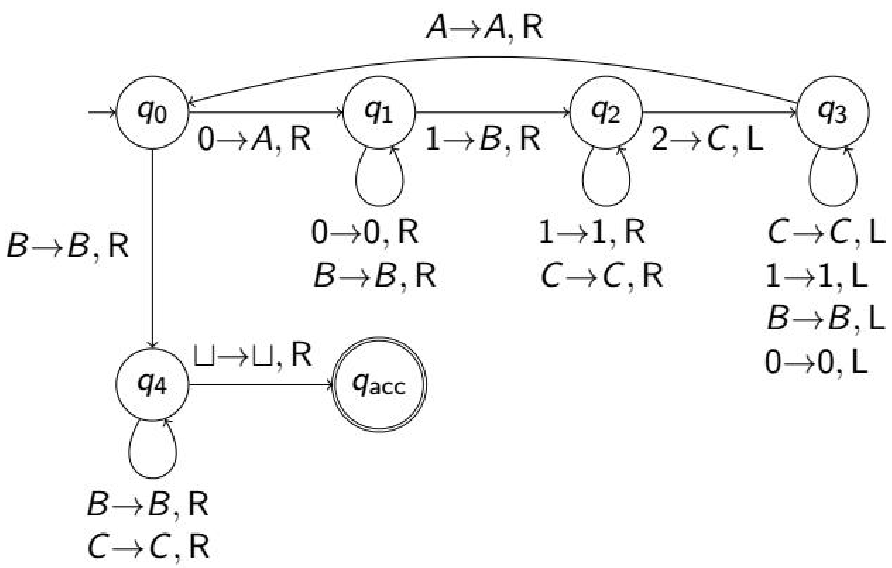
Example 2: TM for $\{0^n1^n2^n \mid n > 0\}$

Design a TM to accept the language $L = \{0^n1^n2^n \mid n > 0\}$

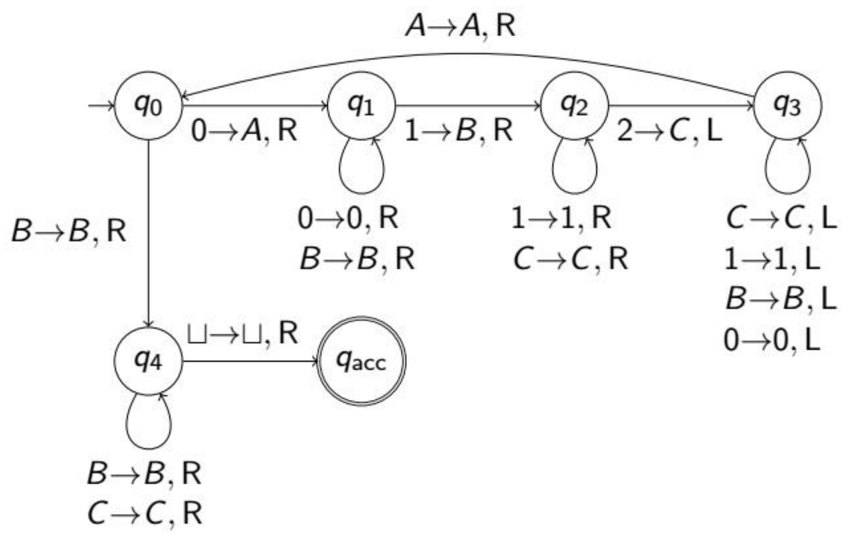
High level description

```
On input string w
    while there are unmarked 0s, do
        Mark the left most 0
        Scan right to reach the leftmost unmarked 1;
            if there is no such 1 then crash
        Mark the leftmost 1
        Scan right to reach the leftmost unmarked 2;
            if there is no such 2 then crash
        Mark the leftmost 2
    done
    Check to see that there are no unmarked 1s or 2s;
        if there are then crash
    accept
```

Example 2: TM for $\{0^n1^n2^n \mid n > 0\}$



Example 2: TM for $\{0^n1^n2^n \mid n > 0\}$



e.g.: $q_0001122 \vdash^* A0Bq_31C2 \vdash^* q_3A0B1C2 \vdash Aq_00B1C2$
$\vdash^* AAq_0BBCC \vdash^* AABBCCq_4\sqcup \vdash AABBCC\sqcup q_{acc}\sqcup$