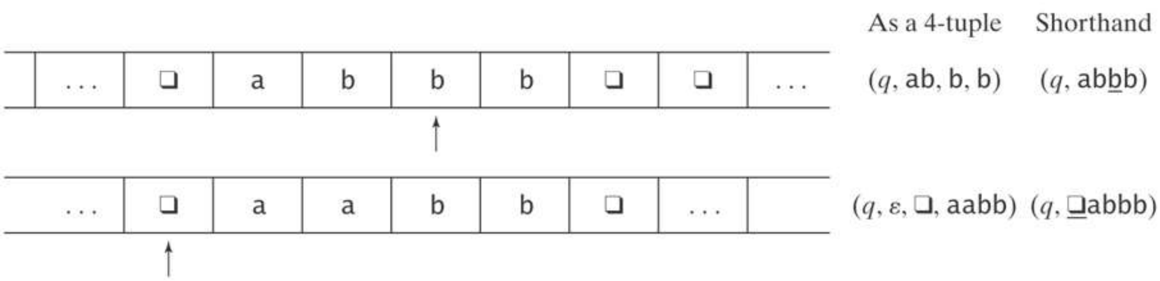
A TM m, is not guaranteed to halt and there exists no algorithm to construct one that is guaranteed to do so.

A TM configuration can be expressed as a 4-tuple: state, tape letters up to scanned square, scanned square, tape letters following scanned square.

Example Configurations

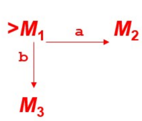


Can combine basic machines to form larger ones.

To do this, we need two forms:

- M_1M_2
  - Begin in start state of M_1, run M_1 until halts, begin M_2 in start state, run M_2 until halts, then halt. If either fails to halt, then M_1M_2 fails to halt.
- $M_1 \xrightarrow{\langle condition \rangle} M_2$
  - The same, except that <*condition*> is checked to move from M_1 to M_2.

Example:



- Start in the start state of M_1. (">" marks the beginning)
- Compute until M_1 reaches a halt state.
- Examine the tape and take the appropriate transition.
- Start in the start state of the next machine, etc.
- Halt if any component reaches a halt state and has no place to go.
- If any component fails to halt, then the entire machine may fail to halt.

SHORTHANDS

Shorthands

$M_1 \underset{b}{\overset{a}{\rightrightarrows}} M_2$	becomes	$M_1 \xrightarrow{\texttt{a, b}} M_2$
$M_1 \xrightarrow{\text{all elems of } \Gamma} M_2$	becomes	$M_1 \longrightarrow M_2$ or $M_1 M_2$

Variables

$M_1 \xrightarrow[\text{except a}]{\text{all elems of } \Gamma} M_2$	becomes and x takes on the value of the current square	$M_1 \xrightarrow{x \leftarrow \neg \texttt{a}} M_2$
$M_1 \xrightarrow{\texttt{a, b}} M_2$	becomes and x takes on the value of the current square	$M_1 \xrightarrow{x \leftarrow \texttt{a, b}} M_2$
	if $x = y$ then take the transition	$M_1 \xrightarrow{x = y} M_2$
e.g., $> \xrightarrow{x \leftarrow \neg \texttt{q}} Rx$	if the current square is not blank, go right and copy it.	

- Symbol writing machines

  For each $x \in \Gamma$, define M_x, written just x, to be a machine that writes x, then halts.

- Head moving machines

  R: for each $x \in \Gamma$, $\delta(s, x) = (s, x, \rightarrow)$
  L: for each $x \in \Gamma$, $\delta(s, x) = (s, x, \leftarrow)$

- Machines that simply halt:

  h, which simply halts.
  n, which halts and rejects.
  y, which halts and accepts.