جامعة الملك سعود
King Saud University

Theory of Computation
CSC 339 – Spring 2021

Chapter-3: part1
Turing Machines

King Saud University

Department of Computer Science

Dr. Azzam Alsudais

Outline

- Recap
- Introduction
- Turing machines

Recap

- **Finite automata (DFA/NFA) are good computational memory when memory is limited.**

Recap

- **Finite automata (DFA/NFA) are good computational memory when memory is limited.**

- **Pushdown automata (PDA) are good computational models when memory is unlimited, but access is limited to a certain order (LIFO)**

Recap

- **Finite automata (DFA/NFA) are good computational memory when memory is limited.**
- **Pushdown automata (PDA) are good computational models when memory is unlimited, but access is limited to a certain order (LIFO)**
  - **Even with the additional memory (stack), PDA are limited.**
    - **Cannot recognize languages such as** $L = \{ww \mid w \in \{0,1\}^*\}$

Recap

- **Finite automata (DFA/NFA) are good computational memory when memory is limited.**
- **Pushdown automata (PDA) are good computational models when memory is unlimited, but access is limited to a certain order (LIFO)**
  - **Even with the additional memory (stack), PDA are limited.**
    - **Cannot recognize languages such as** $L = \{ww \mid w \in \{0,1\}^*\}$

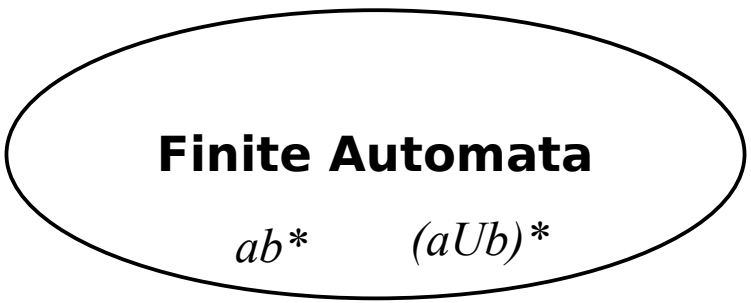
What if we could have unlimited memory

and access it in any way we wish?

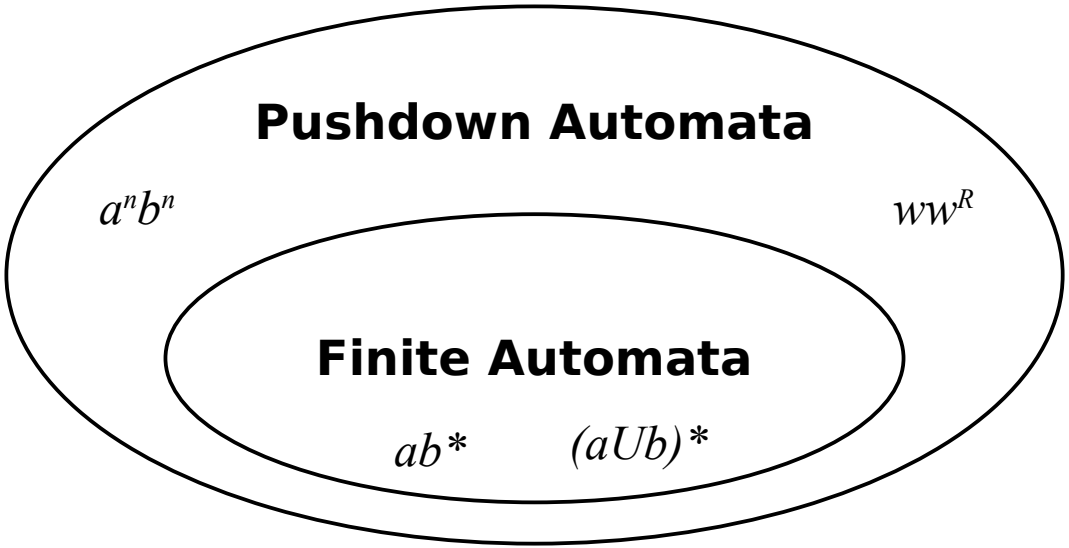
Introduction

- **Turing machines**
  - **Turing machines are more accurate than automata in modeling modern computers.**

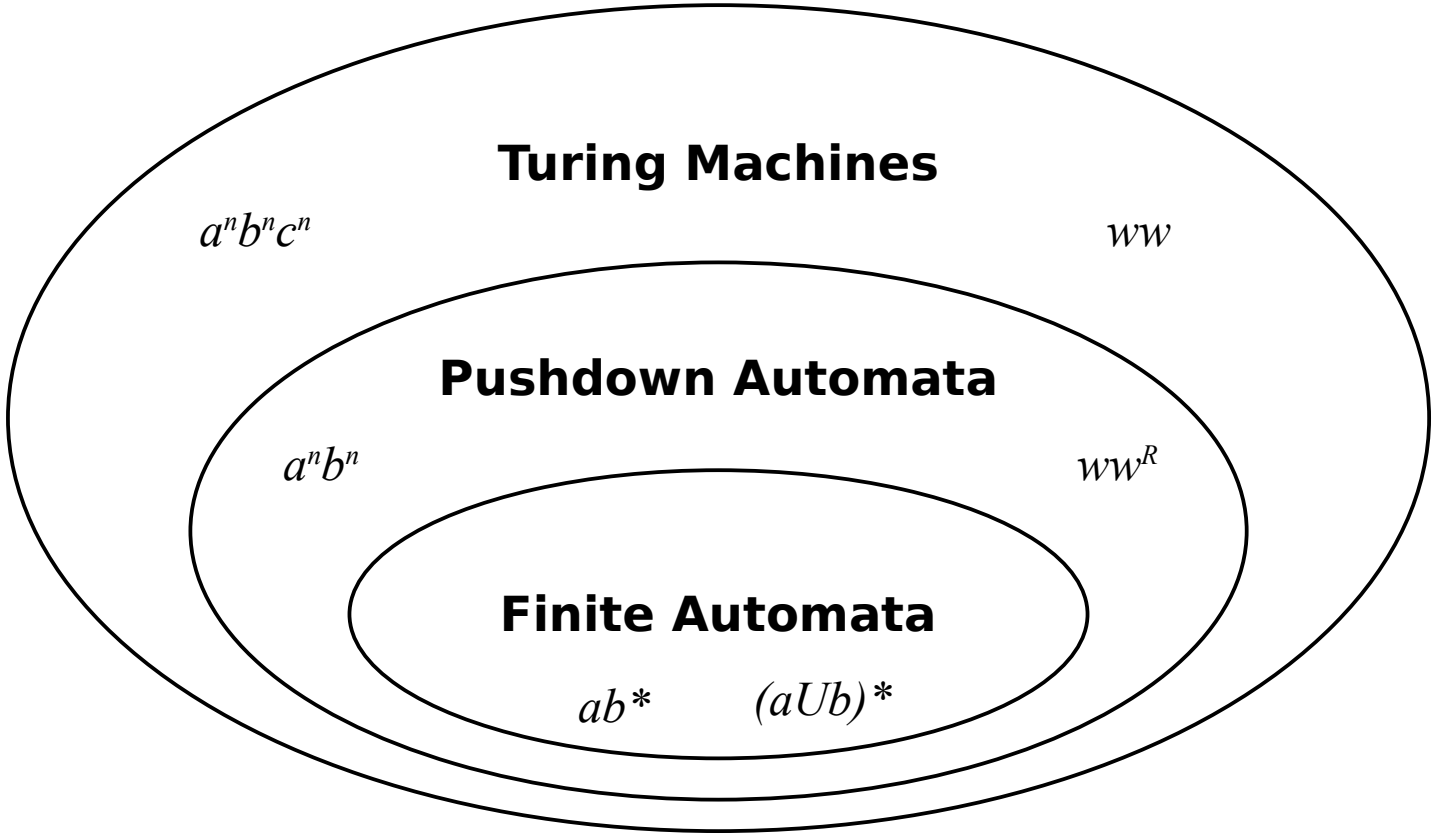
Introduction



Introduction



Introduction

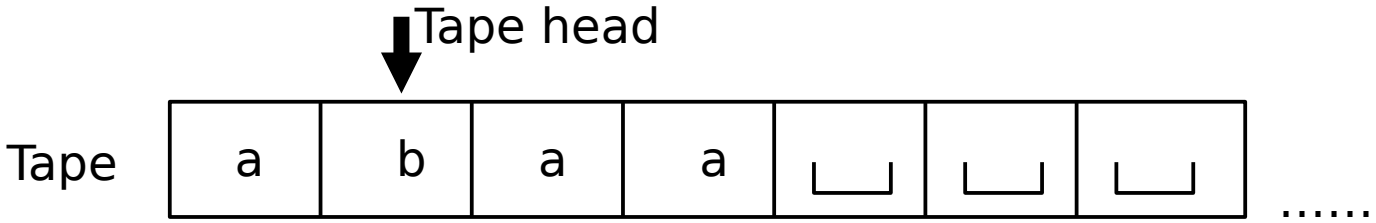


Introduction

- **Turing machines**
  - **Turing machines are more accurate than automata in modeling modern computers.**

Introduction

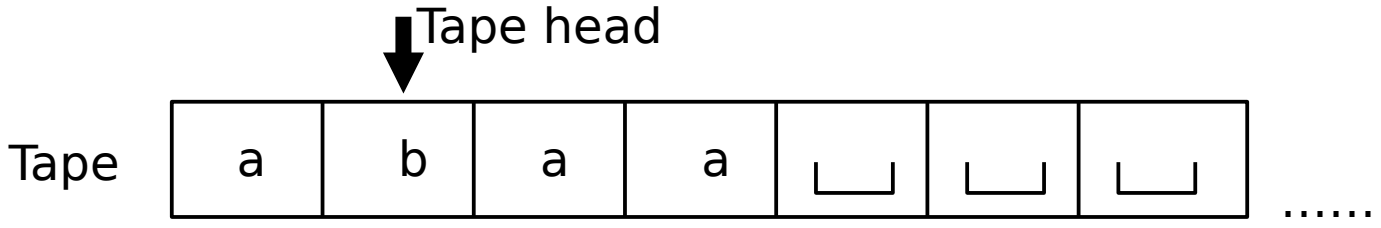
- **Turing machines**
  - **Turing machines are more accurate than automata in modeling modern computers.**
  - **Turing machines have a “tape” that can be used to read and store information.**



Introduction

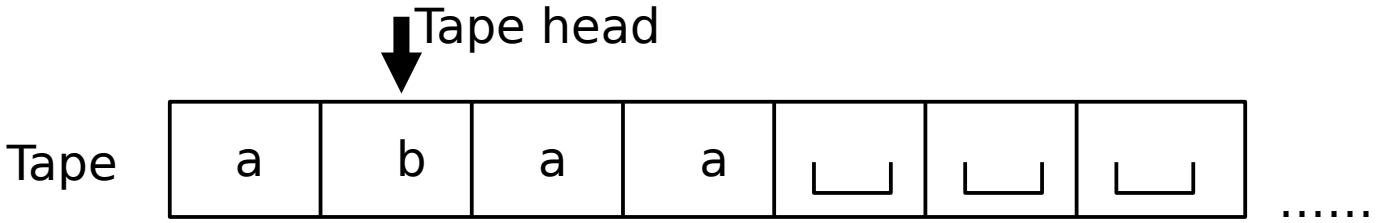
- **Turing machines**
  - **Turing machines are more accurate than automata in modeling modern computers.**
  - **Turing machines have a “tape” that can be used to read and store information.**

Unlimited & unrestricted memory



Introduction

- **Turing machines**
  - **Turing machines are more accurate than automata in modeling modern computers.**
  - **Turing machines have a “tape” that can be used to read and store information.**
  - **The tape initially contains the input string, followed by the blank symbol**



Turing Machines vs. Finite Automata

- Turing machines can read from the tape and write to it.
- The tape “head” can move left and right.
- The tape is infinite.
- Separate accept and reject states.
- The accept and reject states take effect immediately.

Turing Machines: Example

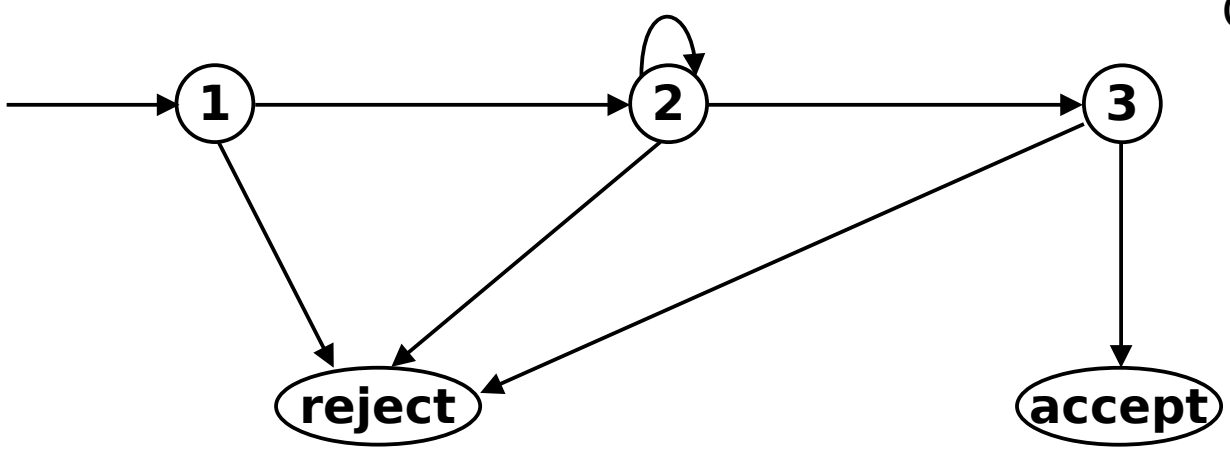
- **Design a turing machine that recognizes L = {ab*a}**

Consider $w = abba$

Turing Machines: Example

- **Design a turing machine that recognizes L = {ab*a}**

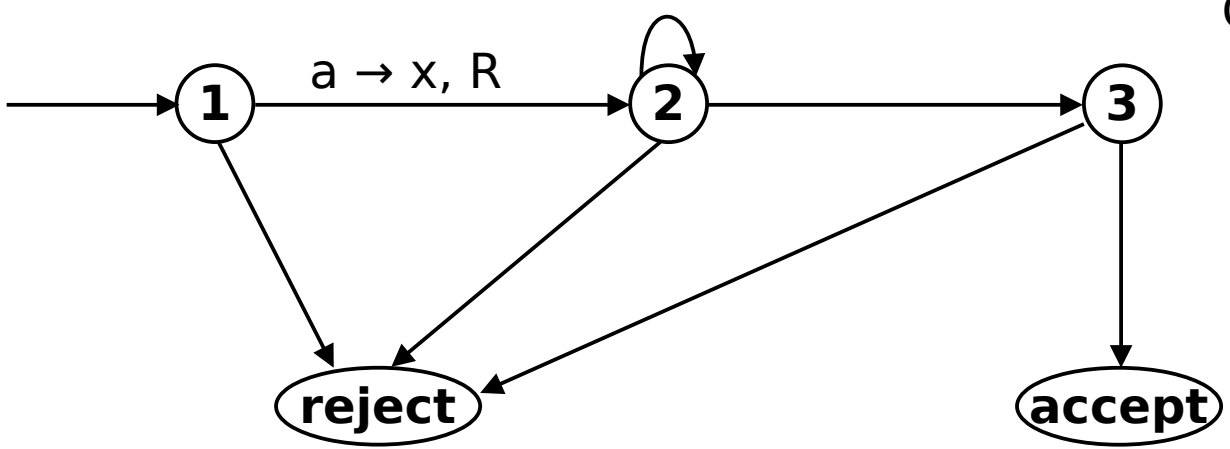
Consider *w* = *abba*



Turing Machines: Example

- **Design a turing machine that recognizes L = {ab*a}**

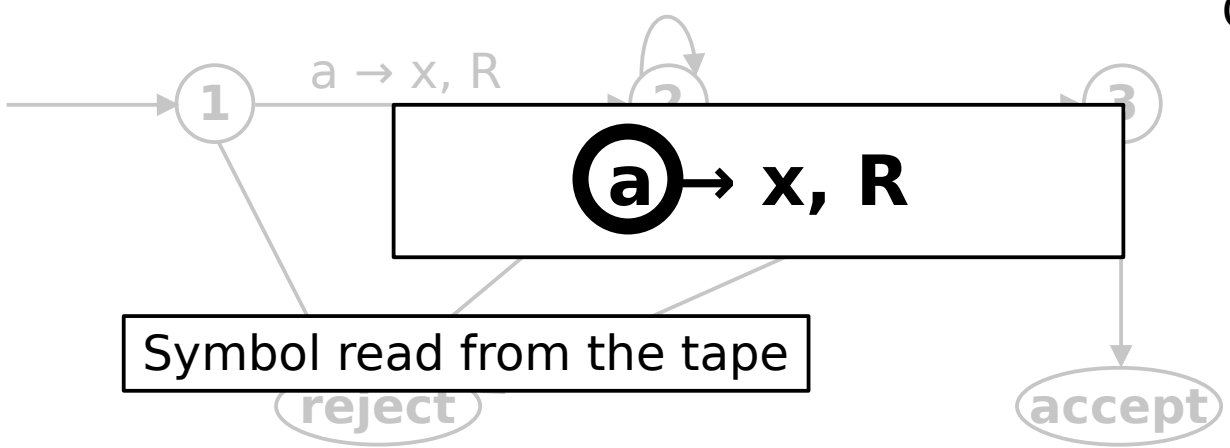
Consider *w* = *abba*



Turing Machines: Example

- **Design a turing machine that recognizes L = {ab*a}**

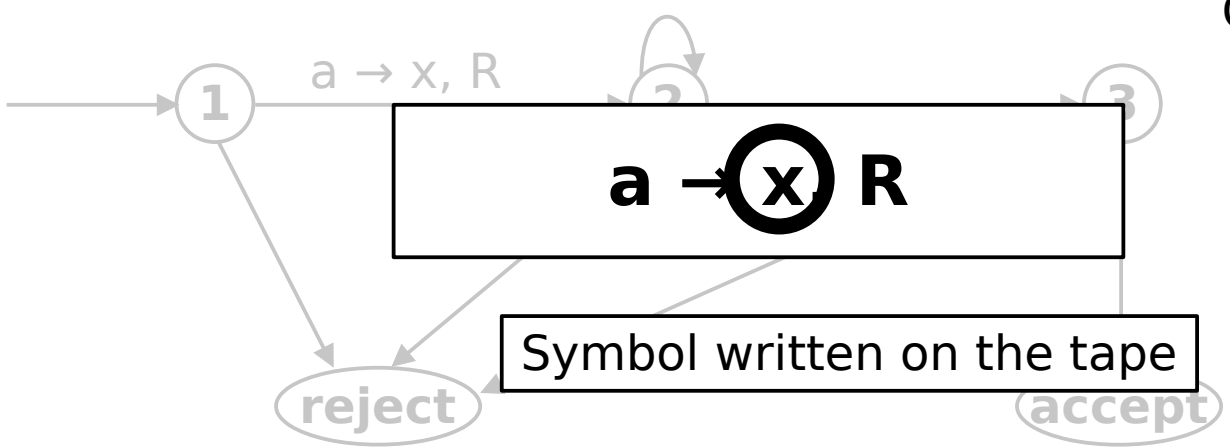
Consider *w* = *abba*



Turing Machines: Example

- **Design a turing machine that recognizes L = {ab*a}**

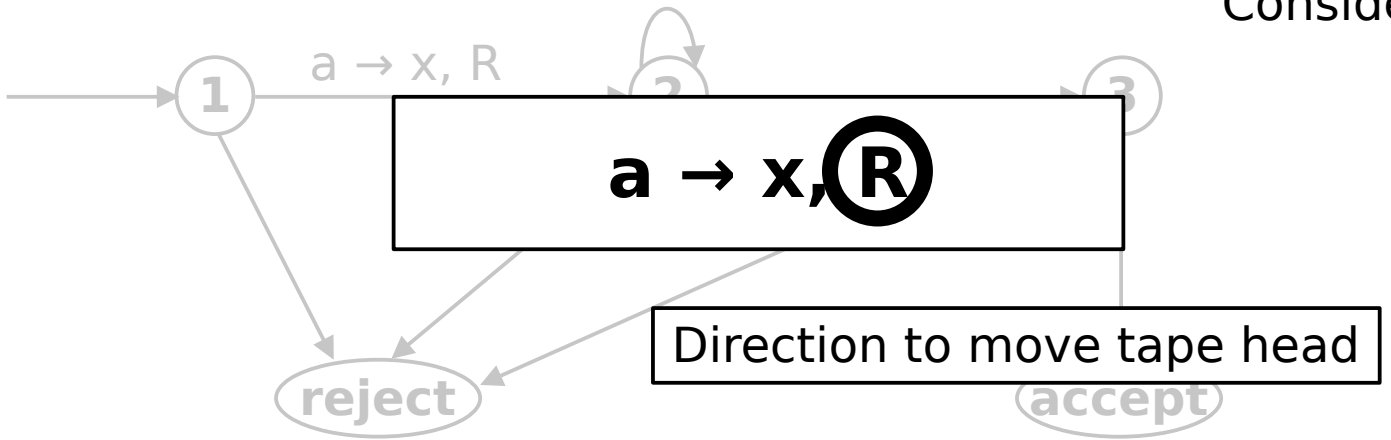
Consider *w* = *abba*



Turing Machines: Example

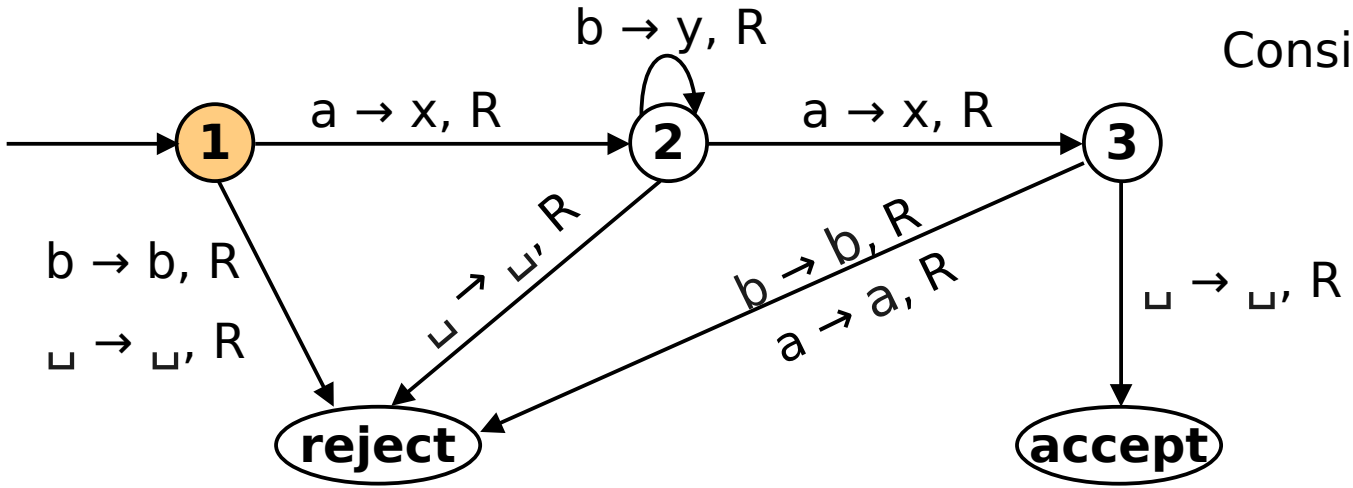
- **Design a turing machine that recognizes L = {ab*a}**

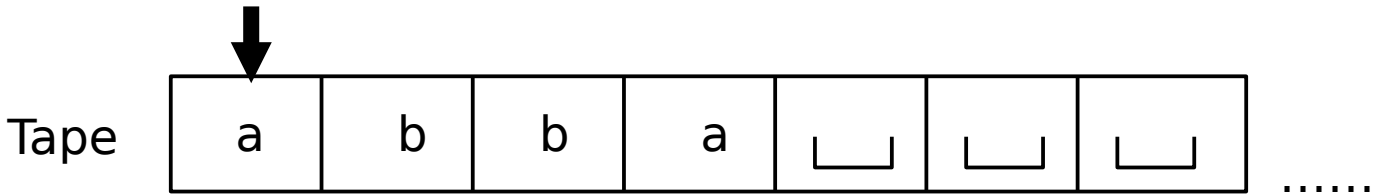
Consider *w* = *abba*



Turing Machines: Example

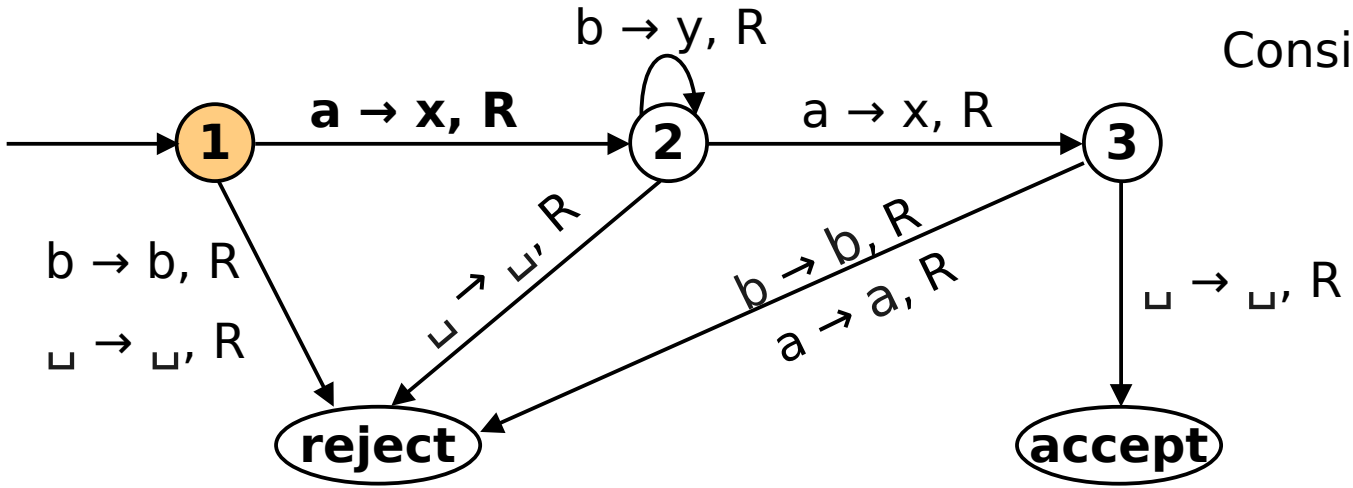
- **Design a turing machine that recognizes L = {ab*a}**

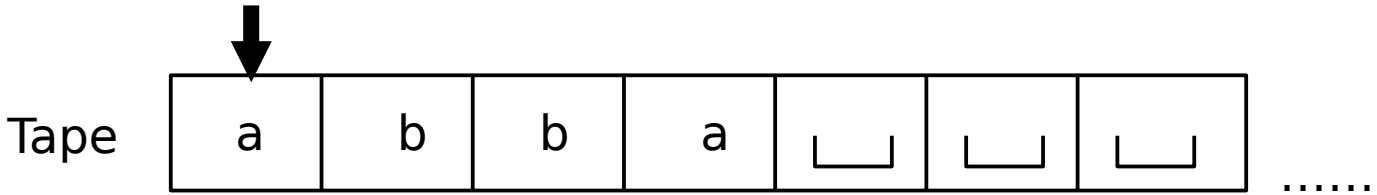




Turing Machines: Example

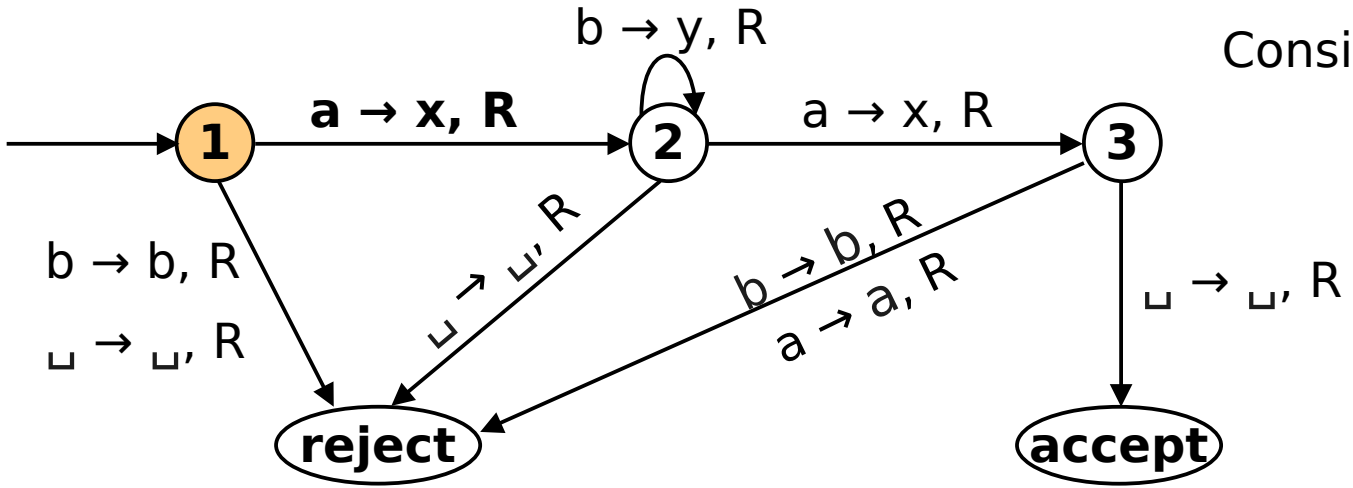
- **Design a turing machine that recognizes L = {ab*a}**

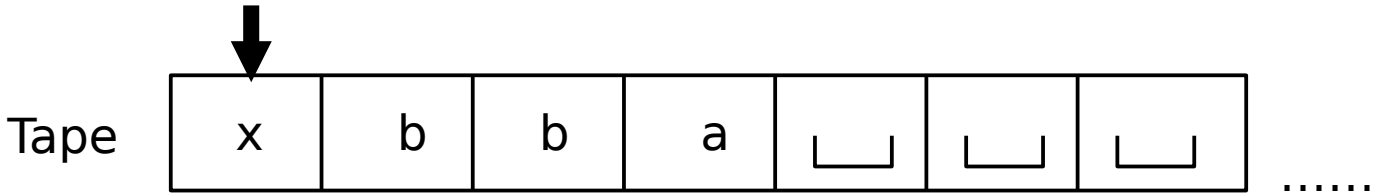




Turing Machines: Example

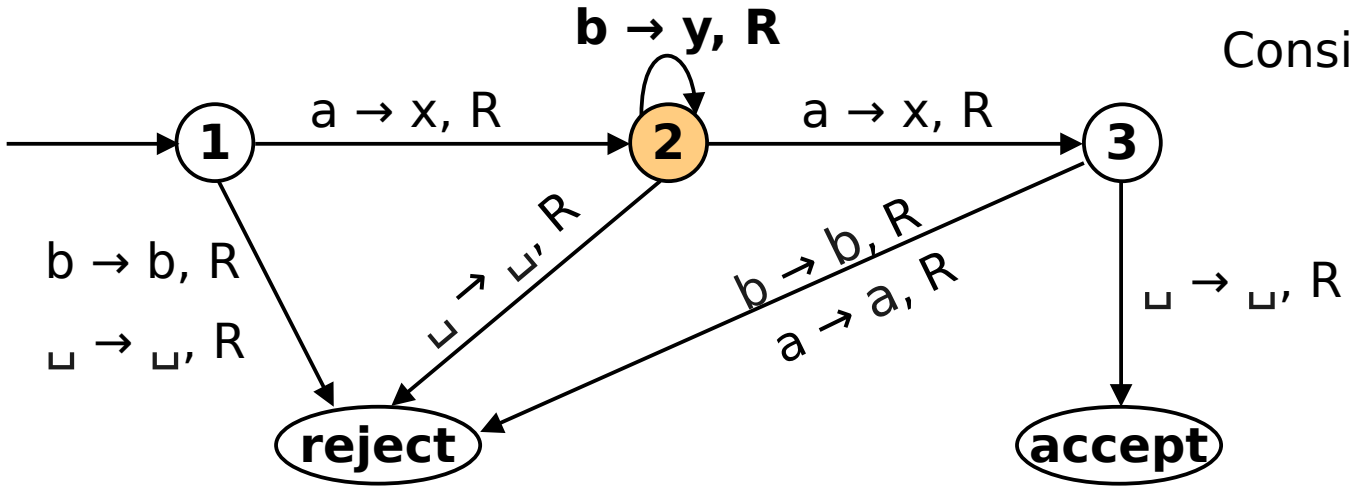
- **Design a turing machine that recognizes L = {ab*a}**





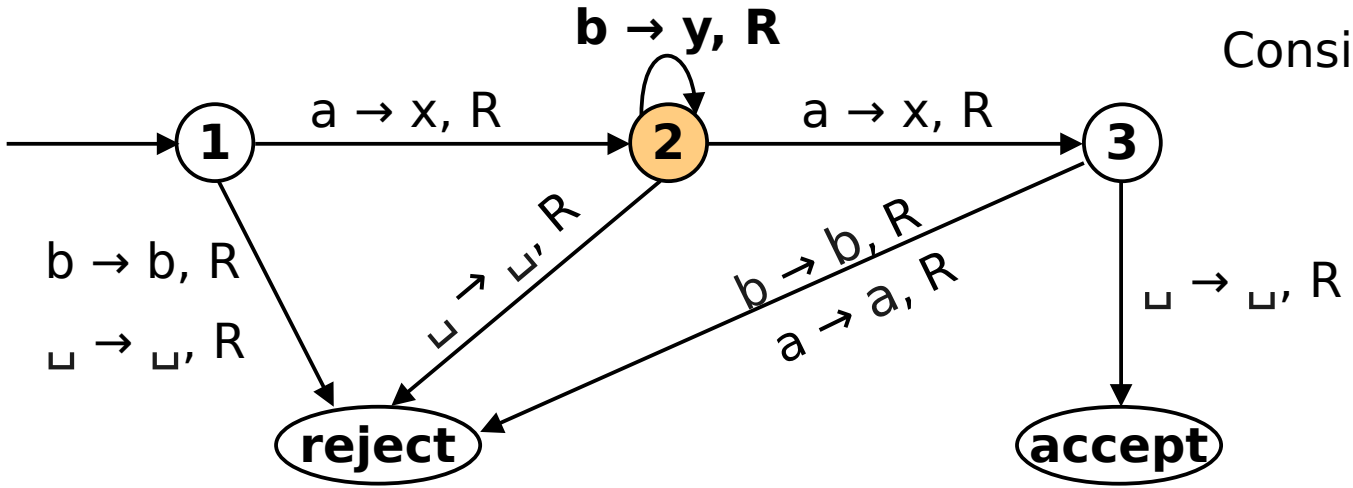
Turing Machines: Example

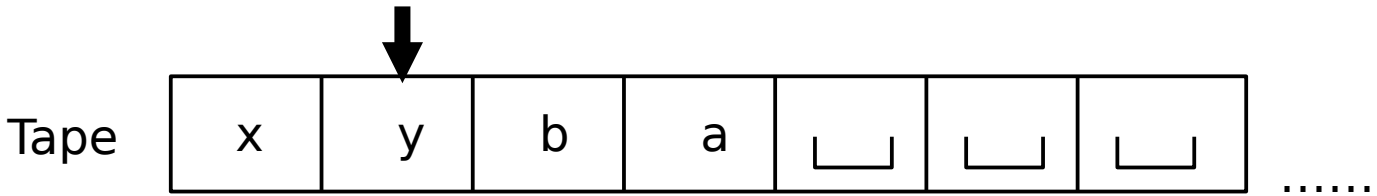
- **Design a turing machine that recognizes L = {ab*a}**



Turing Machines: Example

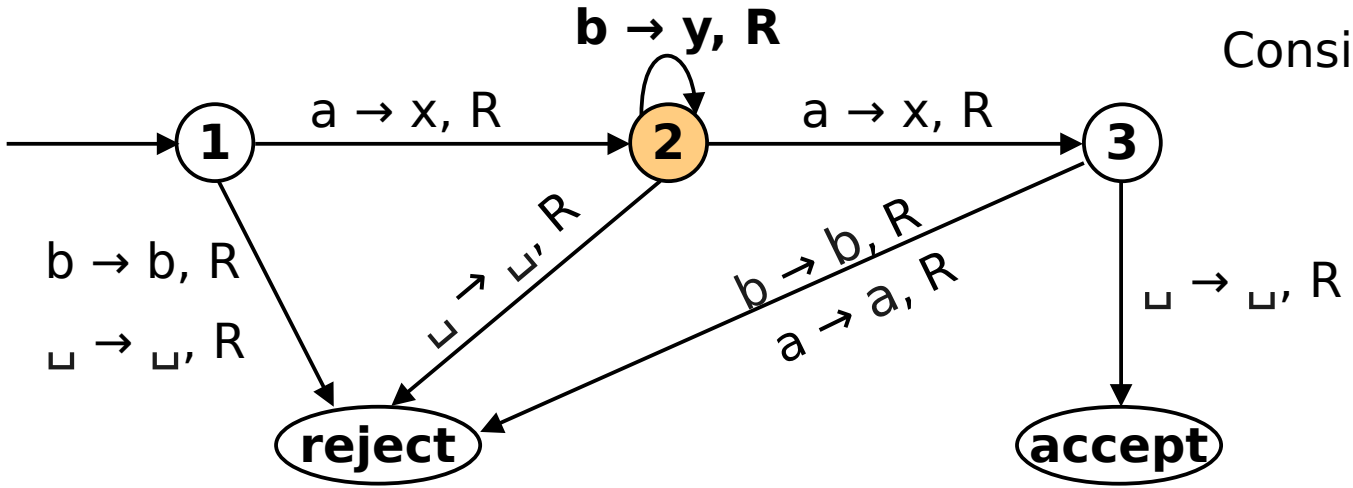
- **Design a turing machine that recognizes L = {ab*a}**

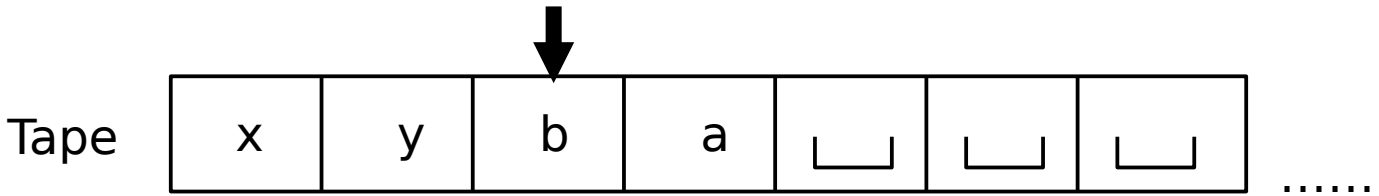




Turing Machines: Example

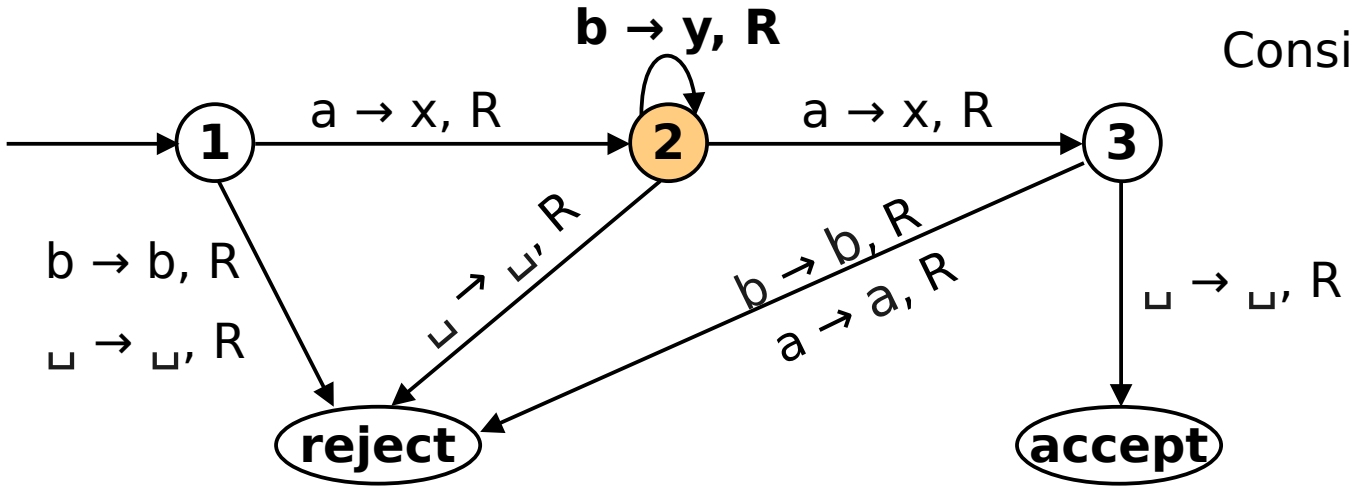
- **Design a turing machine that recognizes L = {ab*a}**

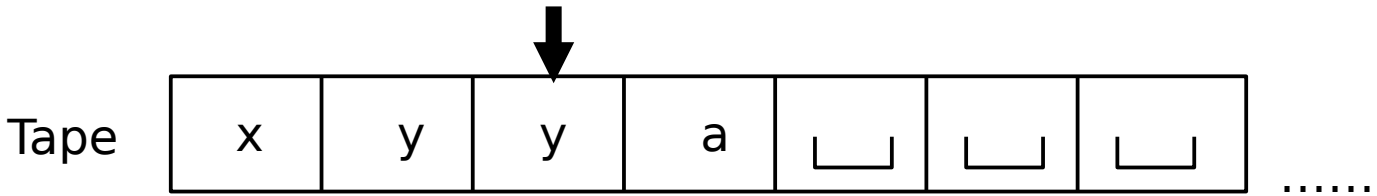




Turing Machines: Example

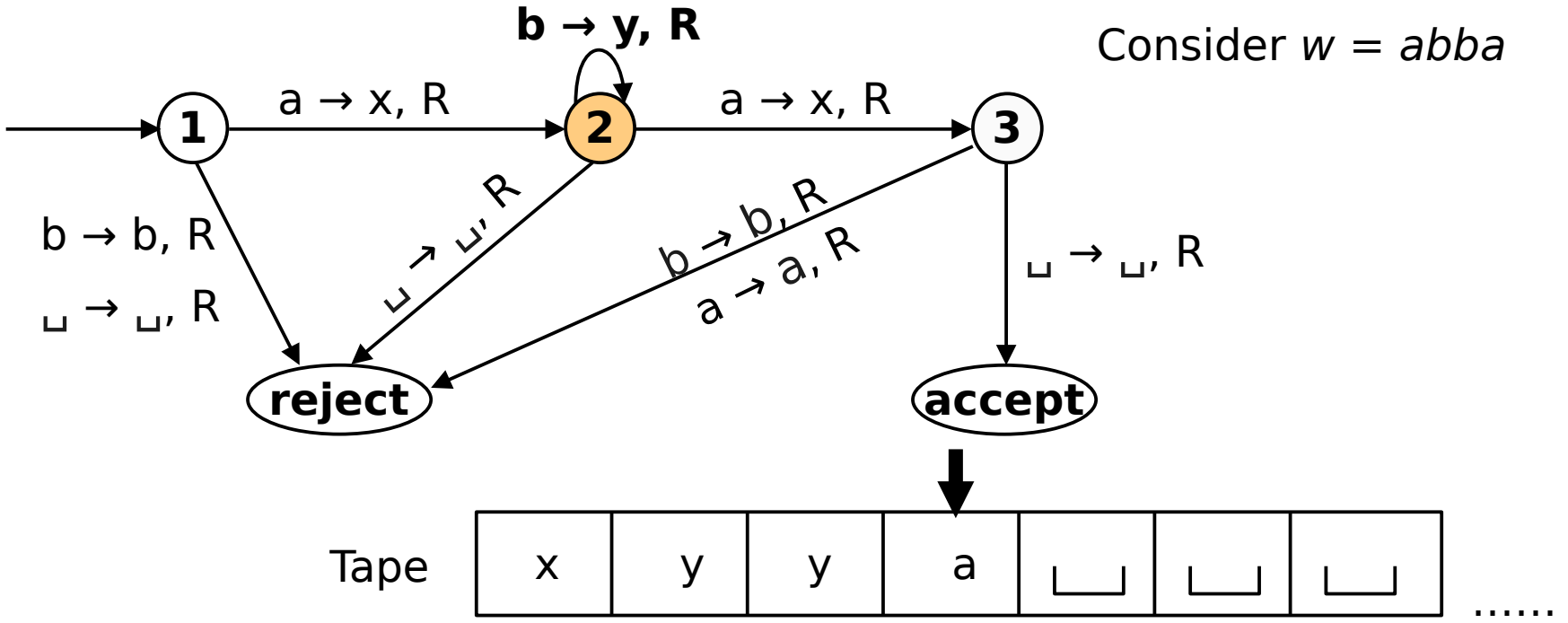
- **Design a turing machine that recognizes L = {ab*a}**





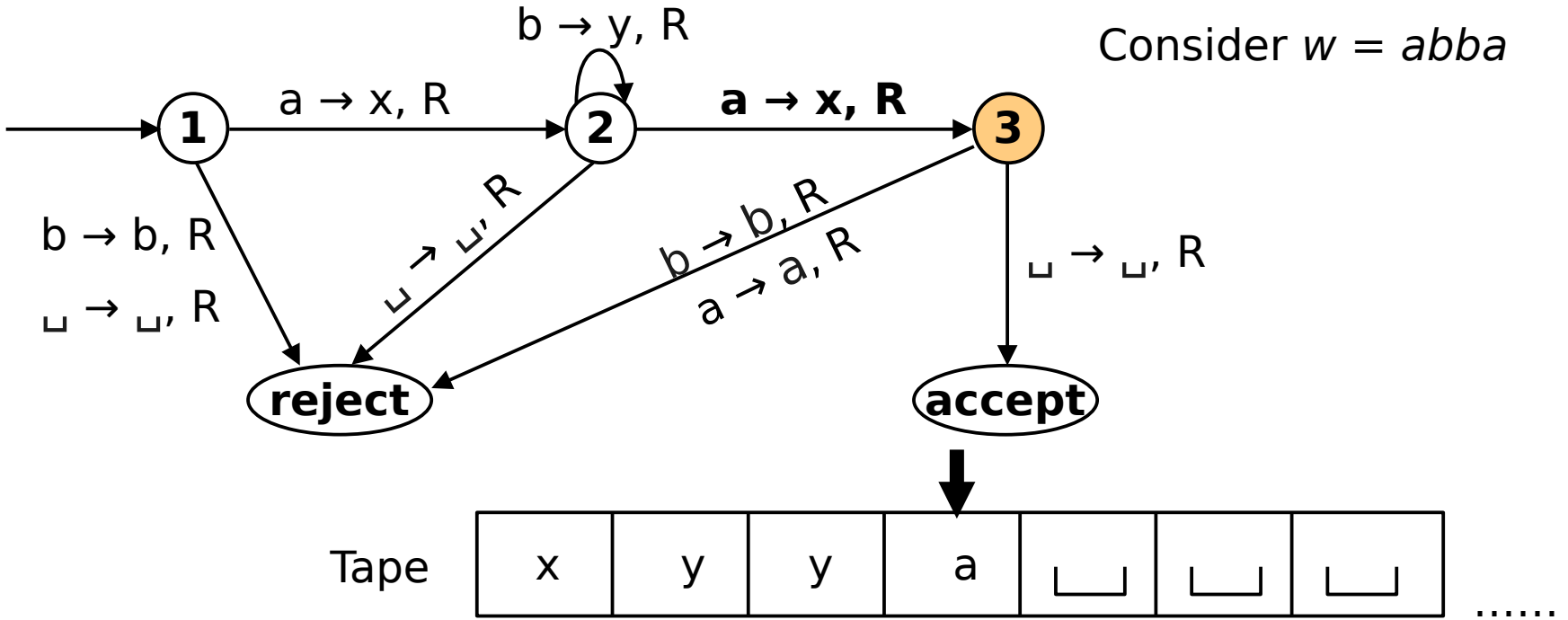
Turing Machines: Example

- **Design a turing machine that recognizes L = {ab*a}**



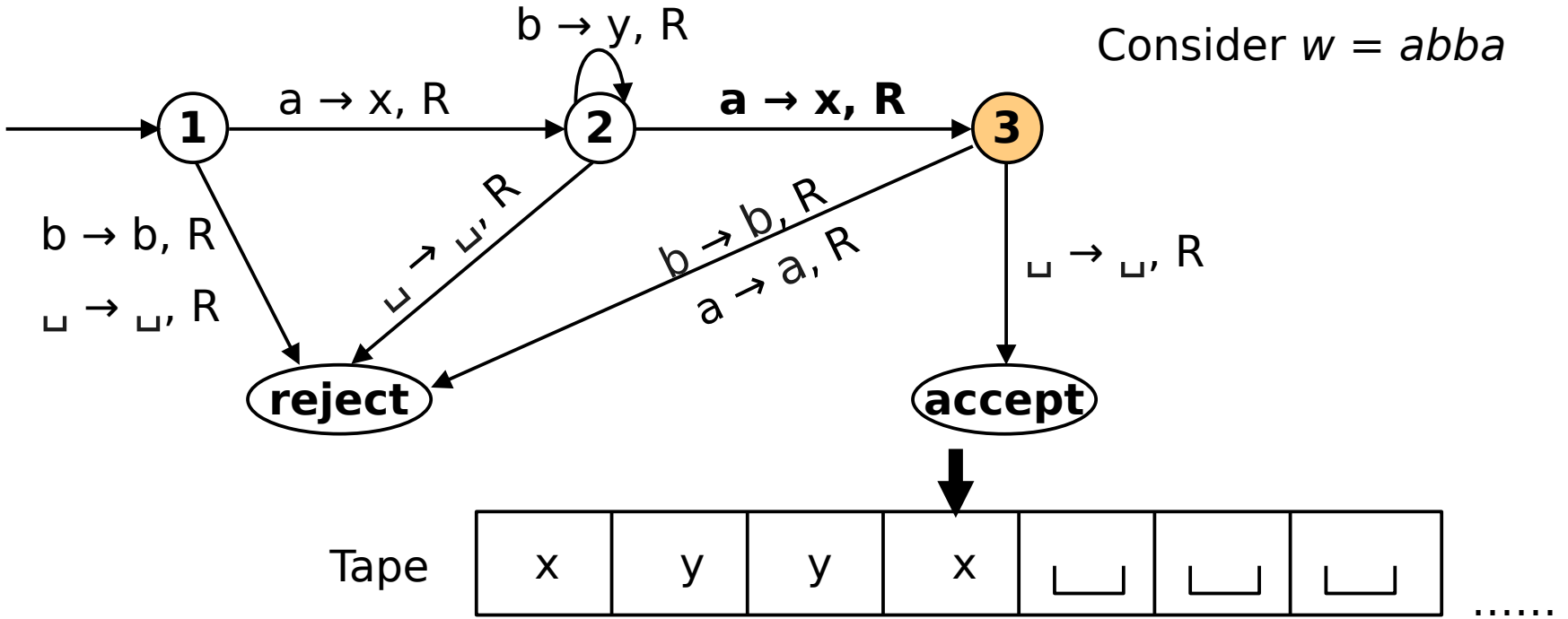
Turing Machines: Example

- **Design a turing machine that recognizes L = {ab*a}**



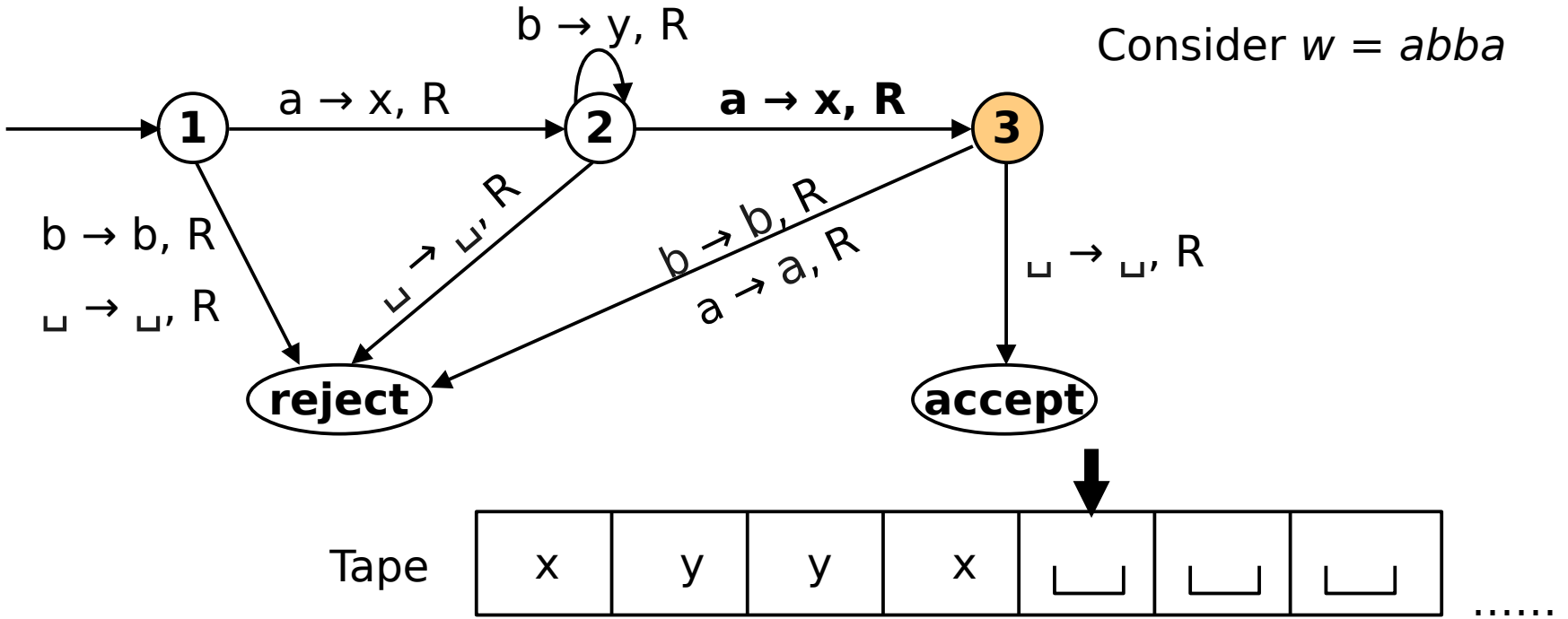
Turing Machines: Example

- **Design a turing machine that recognizes L = {ab*a}**



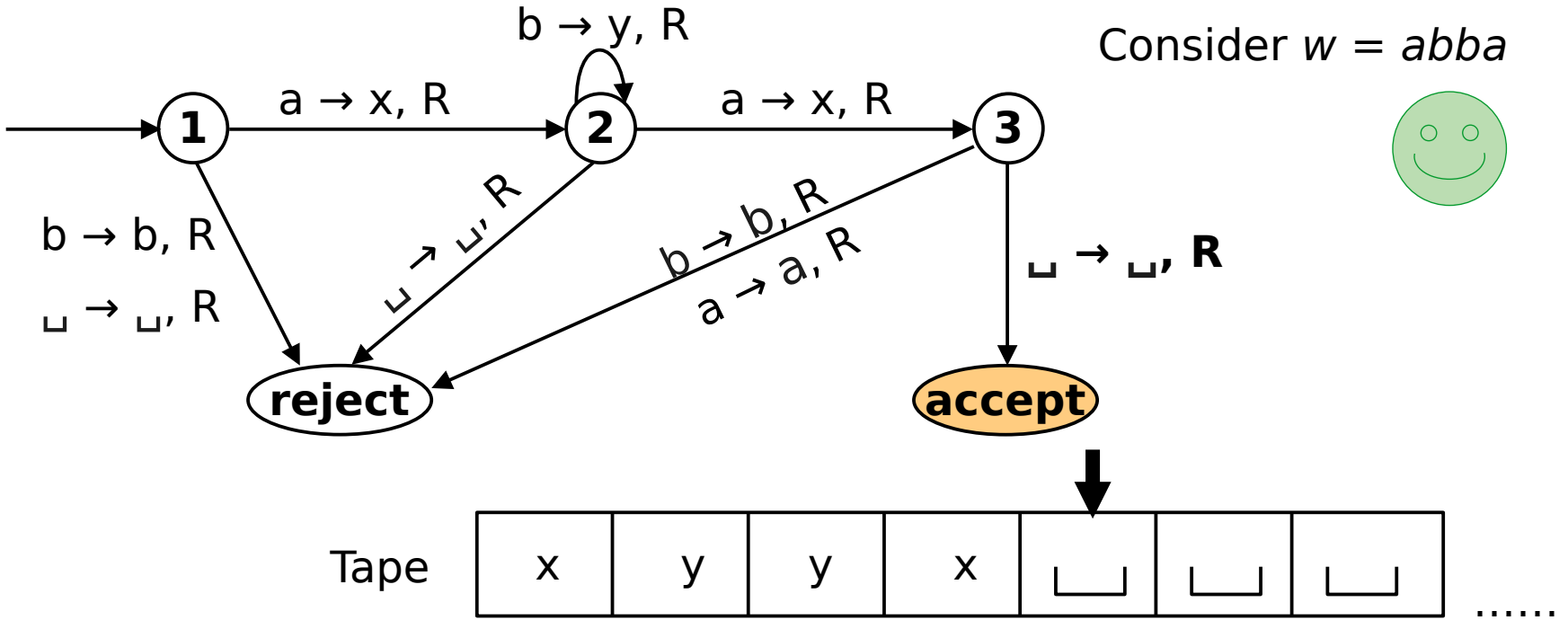
Turing Machines: Example

- **Design a turing machine that recognizes L = {ab*a}**



Turing Machines: Example

- **Design a turing machine that recognizes L = {ab*a}**



Turing Machines: Formal Definition

- A turing machine is a 7-tuple
  - Q is the set of states
  - Σ is the input alphabet not containing the blank symbol ␣ ,
  - Γ is the tape alphabet, where ␣ ∈ Γ and Σ ⊆ Γ,

Turing Machines: Formal Definition

- A turing machine is a 7-tuple
  - Q is the set of states
  - Σ is the input alphabet not containing the blank symbol ␣ ,
  - Γ is the tape alphabet, where ␣ ∈ Γ and Σ ⊆ Γ,
  - $\delta : Q \times \Gamma \rightarrow Q \times \Gamma \times \{L, R\}$ is the transition function,

Turing Machines: Formal Definition

- A turing machine is a 7-tuple
  - Q is the set of states
  - Σ is the input alphabet not containing the blank symbol ␣ ,
  - Γ is the tape alphabet, where ␣ ∈ Γ and Σ ⊆ Γ,
  - $\delta : Q \times \Gamma \rightarrow Q \times \Gamma \times \{L, R\}$ is the transition function,
  - $q_0 \in Q$ is the start state,

Turing Machines: Formal Definition

- A turing machine is a 7-tuple
  - Q is the set of states
  - Σ is the input alphabet not containing the blank symbol ␣ ,
  - Γ is the tape alphabet, where ␣ ∈ Γ and Σ ⊆ Γ,
  - $\delta : Q \times \Gamma \rightarrow Q \times \Gamma \times \{L, R\}$ is the transition function,
  - $q_0 \in Q$ is the start state,
  - $q_{accept} \in Q$ is the accept state, and

Turing Machines: Formal Definition

- A turing machine is a 7-tuple
  - Q is the set of states
  - Σ is the input alphabet not containing the blank symbol ␣ ,
  - Γ is the tape alphabet, where ␣ ∈ Γ and Σ ⊆ Γ,
  - $\delta : Q \times \Gamma \rightarrow Q \times \Gamma \times \{L, R\}$ is the transition function,
  - $q_0 \in Q$ is the start state,
  - $q_{accept} \in Q$ is the accept state, and
  - $q_{reject} \in Q$ is the reject state, where $q_{reject} \neq q_{accept}$.

Turing Machines: Configuration

- **A configuration of a turing machine (TM) describes the current status of the TM at any given time.**

Turing Machines: Configuration

- **A <u>configuration</u> of a turing machine (TM) describes the current status of the TM at any given time.**

$$u\ q\ v$$

Turing Machines: Configuration

- **A <u>configuration</u> of a turing machine (TM) describes the current status of the TM at any given time.**

Current state

Turing Machines: Configuration

- **A <u>configuration</u> of a turing machine (TM) describes the current status of the TM at any given time.**

Current tape content

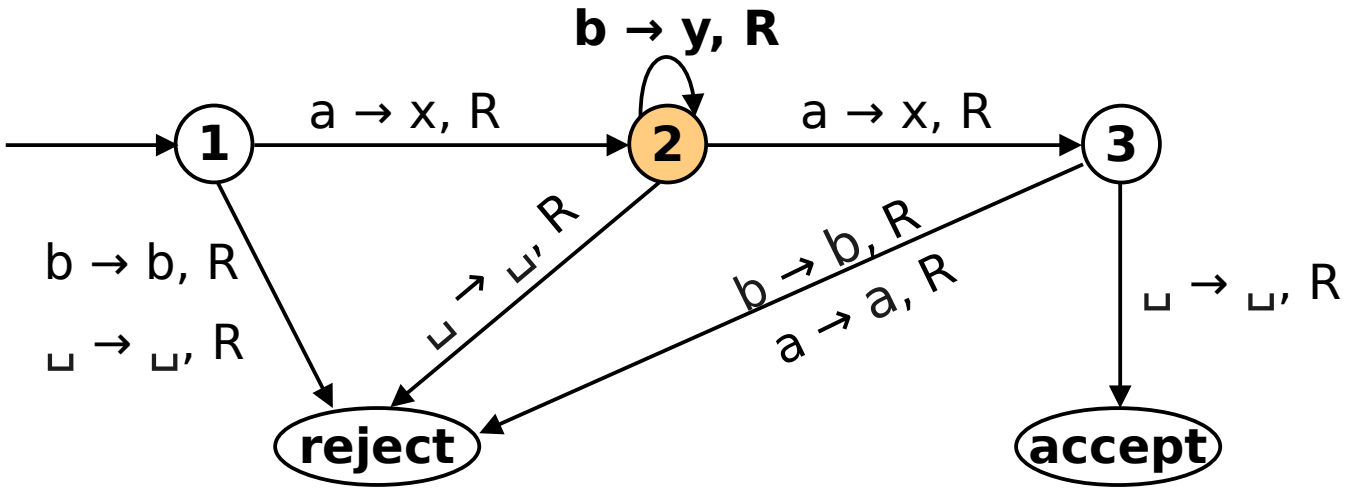
Turing Machines: Configuration

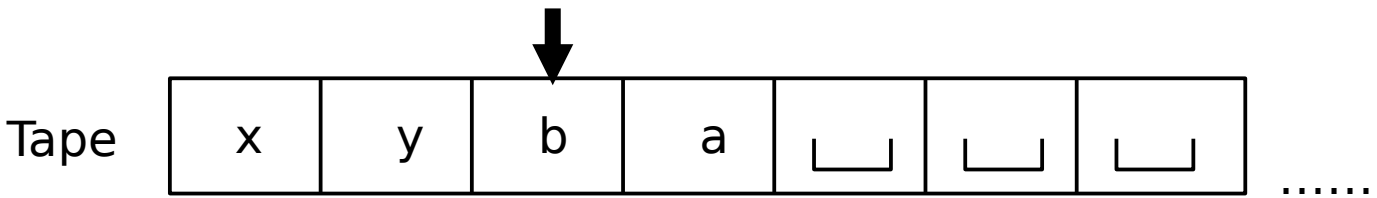
- **A <u>configuration</u> of a turing machine (TM) describes the current status of the TM at any given time.**

$$u\ q(v)$$

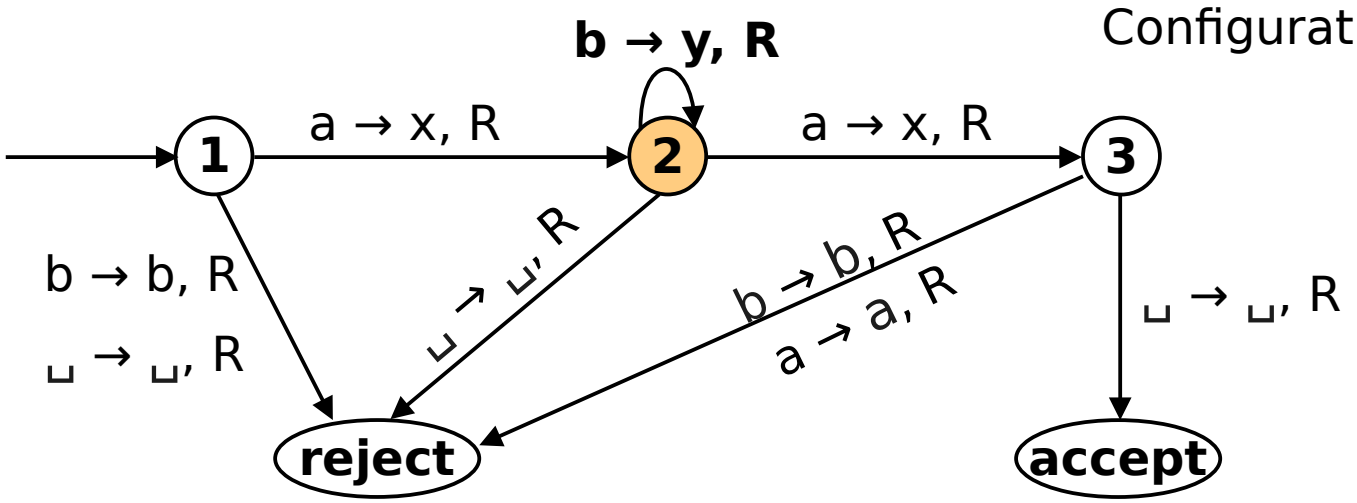
Tape head location is the first symbol of v

Turing Machines: Configuration

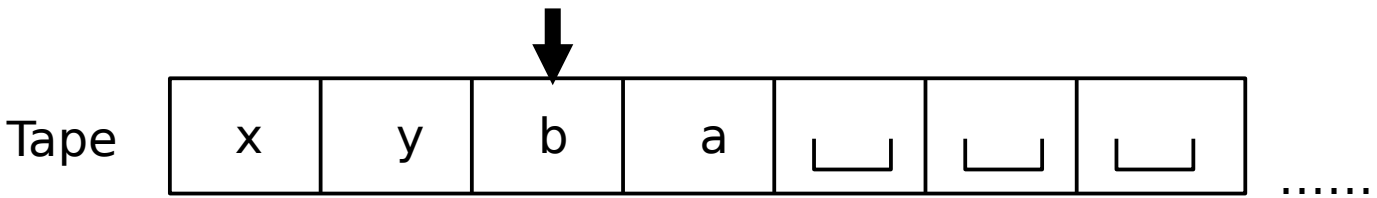




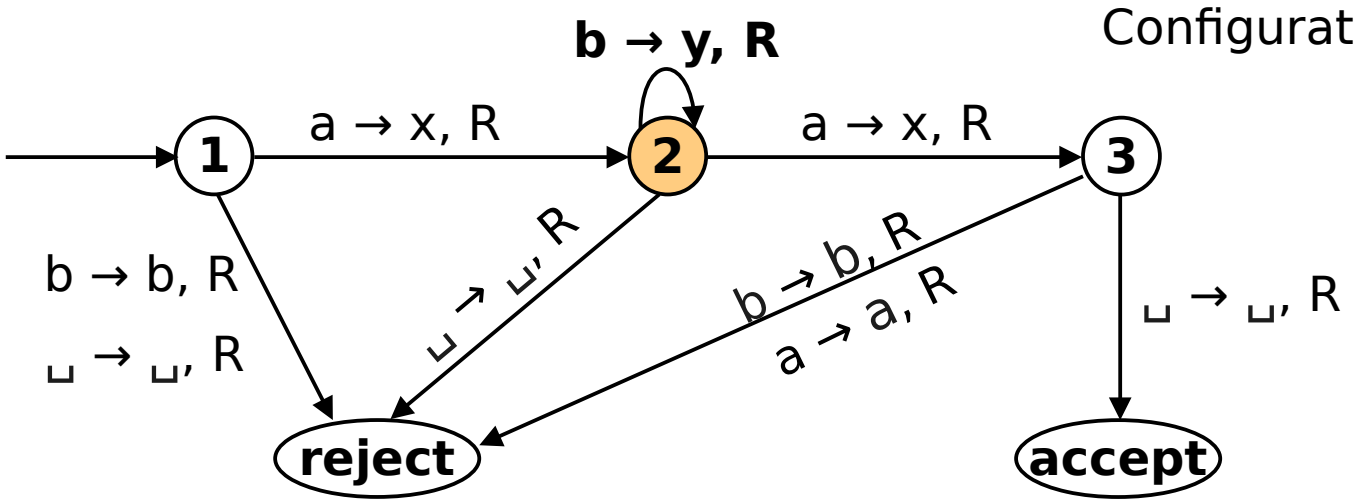
Turing Machines: Configuration



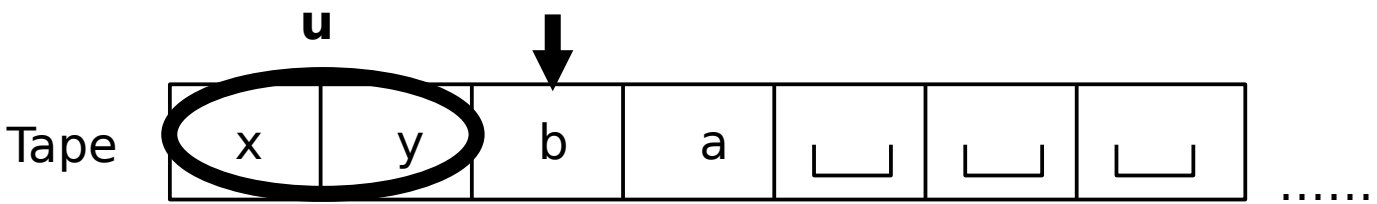
Configuration is **xy2ba**



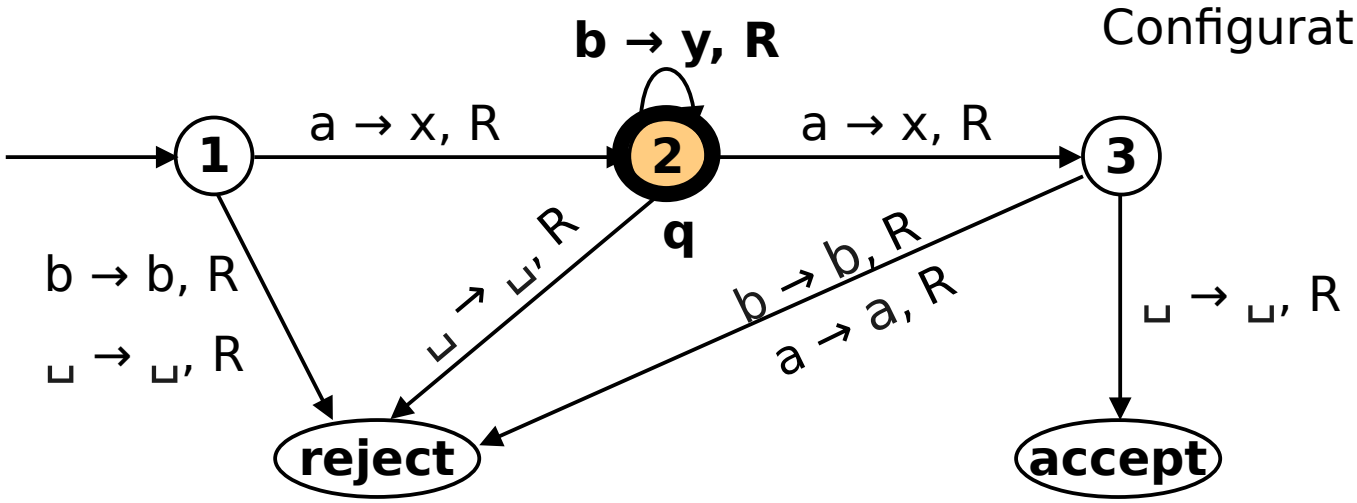
Turing Machines: Configuration



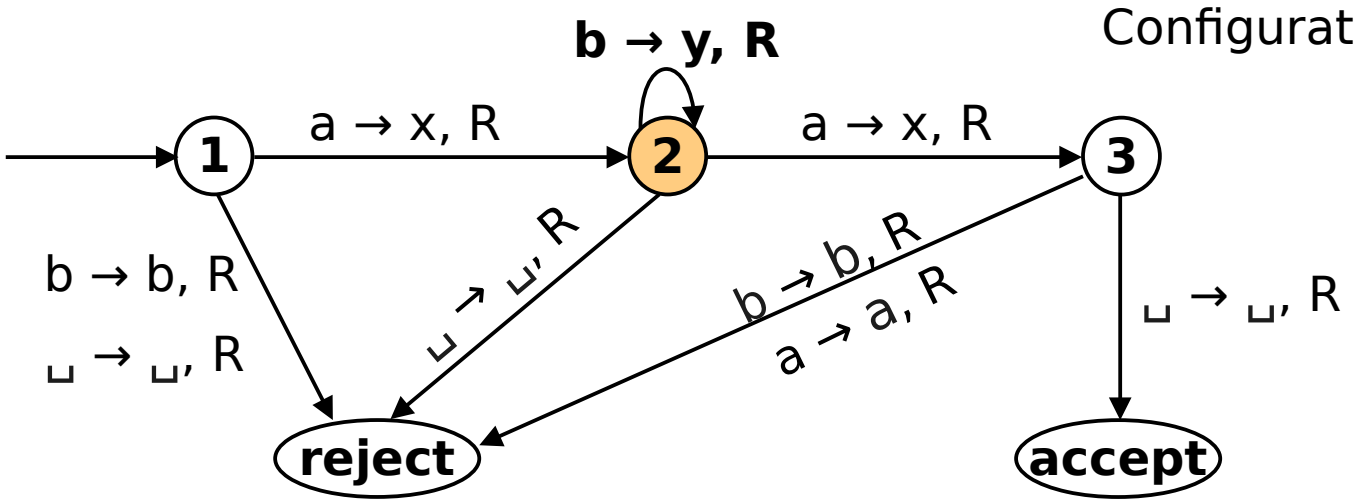
Configuration is **xy2ba**



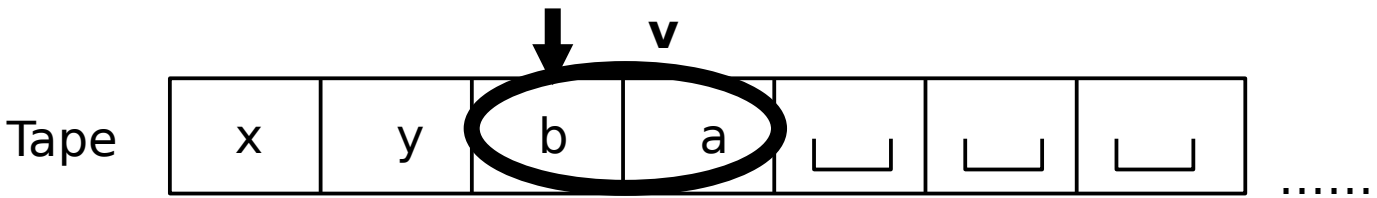
Turing Machines: Configuration



Turing Machines: Configuration



Configuration is **xy2ba**



Turing Machines: Configuration

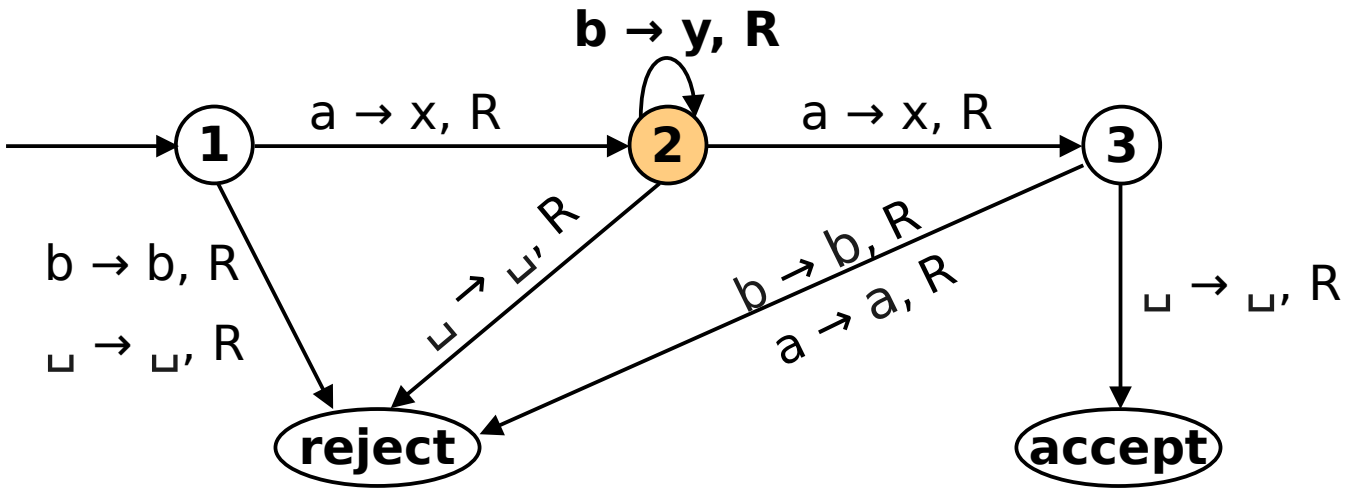
- **Configuration** C_1 **yields** C_2 **if the TM can move from** C_1 **to** C_2 **in a single step.**

Turing Machines: Configuration

C_1 = **xy2ba**

C_2 = **xyy2a**

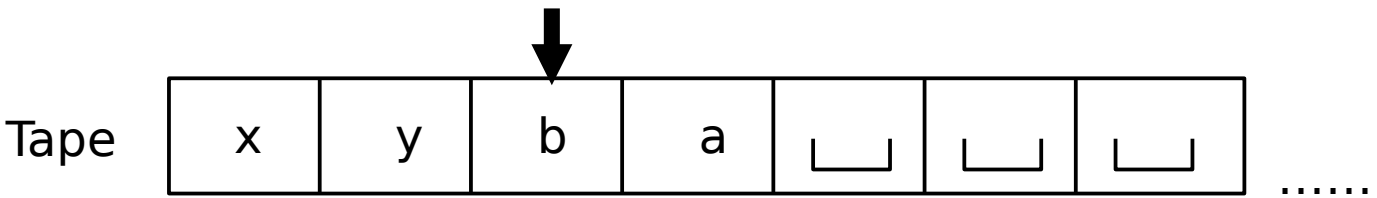
C_3 = **xyyx3**



C_1 yields C_2

C_2 yields C_3

C_1 doesn't yield C_3



Turing Machines: Configuration

- **Start configuration: q_0w, where w is the input string.**

Turing Machines: Configuration

- **Start configuration: q_0w, where w is the input string.**
- **The state in an accepting configuration is q_{accept}.**

Turing Machines: Configuration

- **Start configuration: q_0w, where w is the input string.**
- **The state in an accepting configuration is q_{accept}.**
- **The state in a rejecting configuration is q_{reject}.**

Turing Machines: Configuration

- **Start configuration: q_0w, where w is the input string.**
- **The state in an accepting configuration is q_{accept}.**
- **The state in a rejecting configuration is q_{reject}.**

Halting configurations

Turing Machines: Configuration

- **A TM M accepts its input w if there is a sequence of configurations $C_1, C_2, \ldots, C_k$ exists, where:**
  - **C_1 is the start configuration of M on input w.**
  - **each C_i yields C_{i+1}, and**
  - **C_k is an accepting configuration.**

Turing Machines: Turing-recognizable

- **The collection of strings that *M* accepts is the language of *M*, or language recognized by *M*, *L(M)*.**
  - *A language is called **Turing-recognizable** if some TM recognizes it.*

Turing Machines: Turing-decidable

- When a TM starts on an input, there are three possible outcomes. The machine may:
  - Accept,
  - Reject, or
  - Loop

Turing Machines: Turing-decidable

- **Machines that loop forever never halt, and thus, never decide whether the input is accepted or rejected.**

Turing Machines: Turing-decidable

- **Machines that loop forever never halt, and thus, never decide whether the input is accepted or rejected.**

- **When a machine always halts (accept or reject), we call that machine a *decider*.**

Turing Machines: Turing-decidable

- **Machines that loop forever never halt, and thus, never decide whether the input is accepted or rejected.**

- **When a machine always halts (accept or reject), we call that machine a *decider*.**

> *A language **Turing-decidable** or simply decidable if some Turing machine decides it*

Turing Machines: Turing-decidable

- **Machines that loop forever never halt, and thus, never decide whether the input is accepted or rejected.**

- **When a machine always halts (accept or reject), we call that machine a *decider*.**

> *A language **Turing-decidable** or simply decidable if some Turing machine decides it*

- **Every Turing-decidable language is also Turing-recognizable.**
  - **But, not the opposite!**

Turing Machines: Example

$$L(M_1) = \{w\#w \mid w \in \{0,1\}^*\}$$

Turing Machines: Example

$$L(M_1) = \{w\#w \mid w \in \{0,1\}^*\}$$

How can we design a Turing machine?

Turing Machines: Example

$L(M_1) = \{w\#w \mid w \in \{0,1\}^*\}$

How can we design a Turing machine?

Think of it as an algorithm

Turing Machines: Example

$L(M_1) = \{w\#w \mid w \in \{0,1\}^*\}$

How can we design a Turing machine?

Think of it as an algorithm

What are the instructions that need to be executed in order to recognize a certain string?

Turing Machines: Example

$L(M_1) = \{w\#w \mid w \in \{0,1\}^*\}$

How can we design a Turing machine?

Think of it as an algorithm

Use the algorithm as a guide to design the TM

Turing Machines: Example

$$L(M_1) = \{w\#w \mid w \in \{0,1\}^*\}$$

Algorithm for $\mathbf{M_1}$

1) Scan the first leftmost input symbol on the tape, and cross it off (replace with x).

Turing Machines: Example

$$L(M_1) = \{w\#w \mid w \in \{0,1\}^*\}$$

Algorithm for $\mathbf{M_1}$

1) Scan the first leftmost input symbol on the tape, and cross it off (replace with x).
2) Move to the first input symbol after #, and cross it off if it's the same symbol that we found in step 1. Else, reject.

Turing Machines: Example

$$L(M_1) = \{w\#w \mid w \in \{0,1\}^*\}$$

Algorithm for $\mathbf{M_1}$

1) Scan the first leftmost input symbol on the tape, and cross it off (replace with x).
2) Move to the first input symbol after #, and cross it off if it's the same symbol that we found in step 1. Else, reject.
3) Repeat steps 1 and 2 until all symbols left to # are crossed off.

Turing Machines: Example

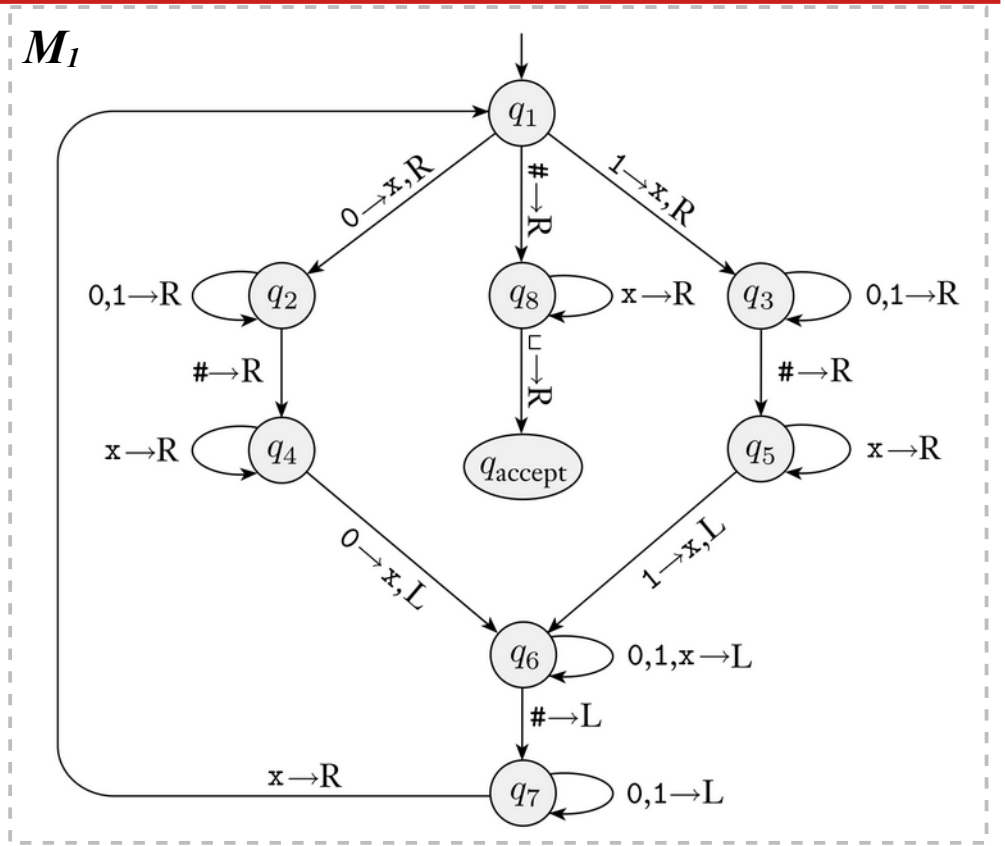
$$L(M_1) = \{w\#w \mid w \in \{0,1\}^*\}$$

Algorithm for M_1

1) Scan the first leftmost input symbol on the tape, and cross it off (replace with x).
2) Move to the first input symbol after #, and cross it off if it's the same symbol that we found in step 1. Else, reject.
3) Repeat steps 1 and 2 until all symbols left to # are crossed off.
4) Scan symbols right to #, and reject if any symbols remain. Else, accept.

Turing Machines: Example

$$L(M_1) = \{w\#w \mid w \in \{0,1\}^*\}$$

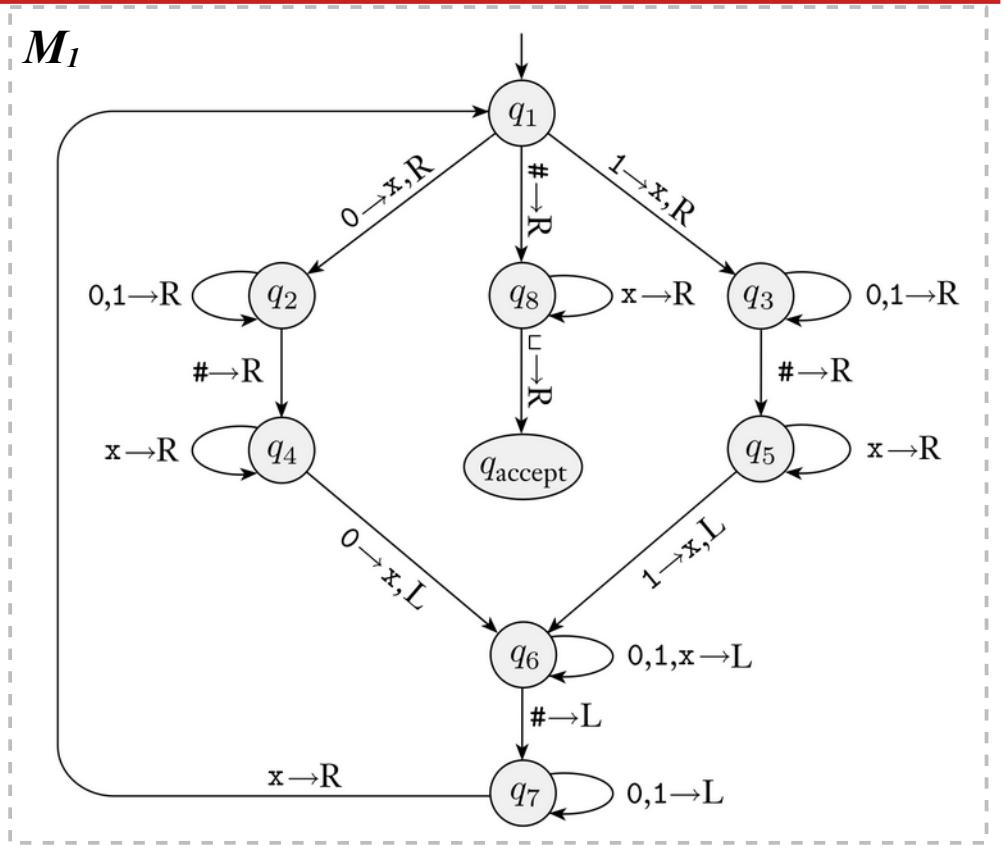


Turing Machines: Example

$$L(M_1) = \{w\#w \mid w \in \{0,1\}^*\}$$

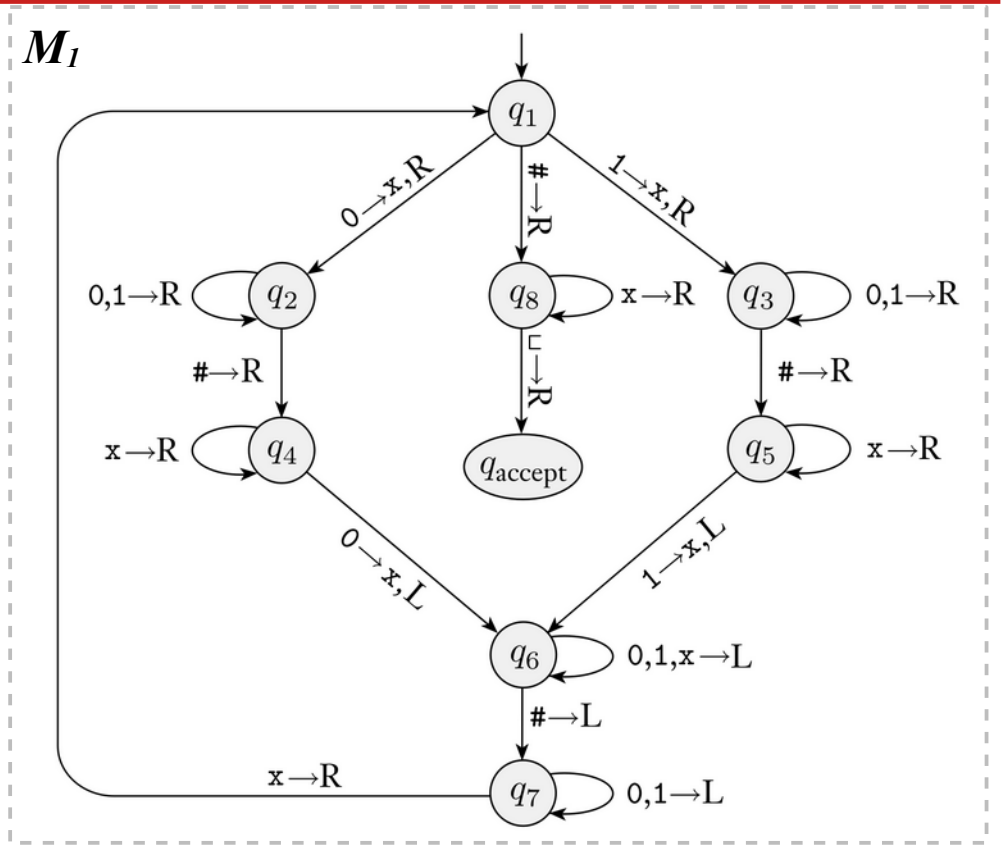
Definition of M_1

- $Q = \{q_1, \ldots, q_8, q_{accept}, q_{reject}\}$
- $\Sigma = \{0,1,\#\}$, and
- $\Gamma = \{0,1,\#,x,\sqcup\}$
- δ is the state diagram
- Start state is q_1
- Accept state is q_{accept}
- Reject state is q_{reject} (not shown)



Turing Machines: Example

Consider $w = 001\#001$

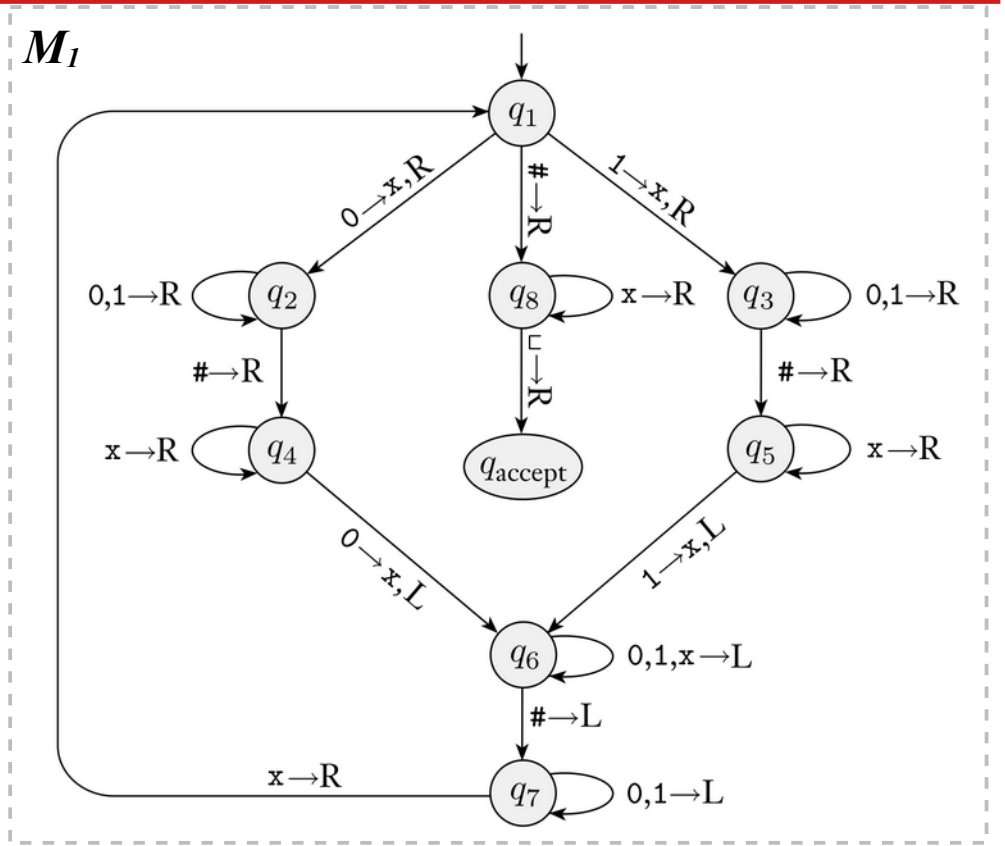


Turing Machines: Example

Consider $w = 001\#001$

$q_1001\#001\sqcup$

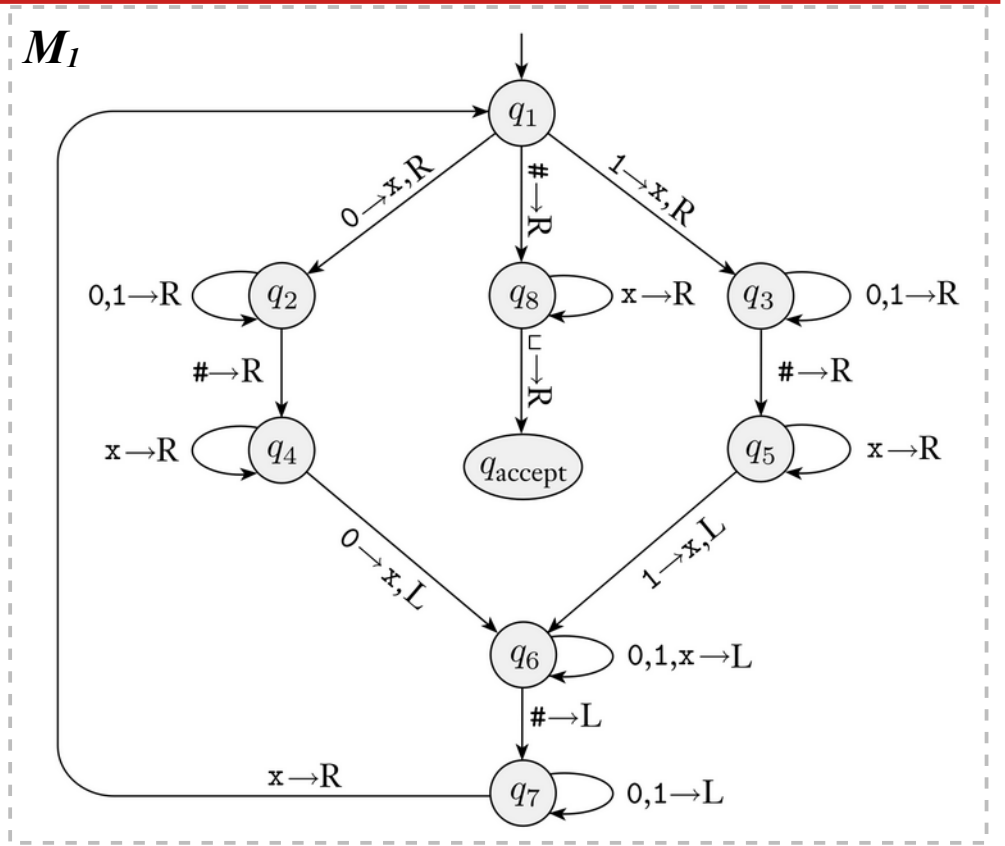
Start configuration



Turing Machines: Example

Consider $w = 001\#001$

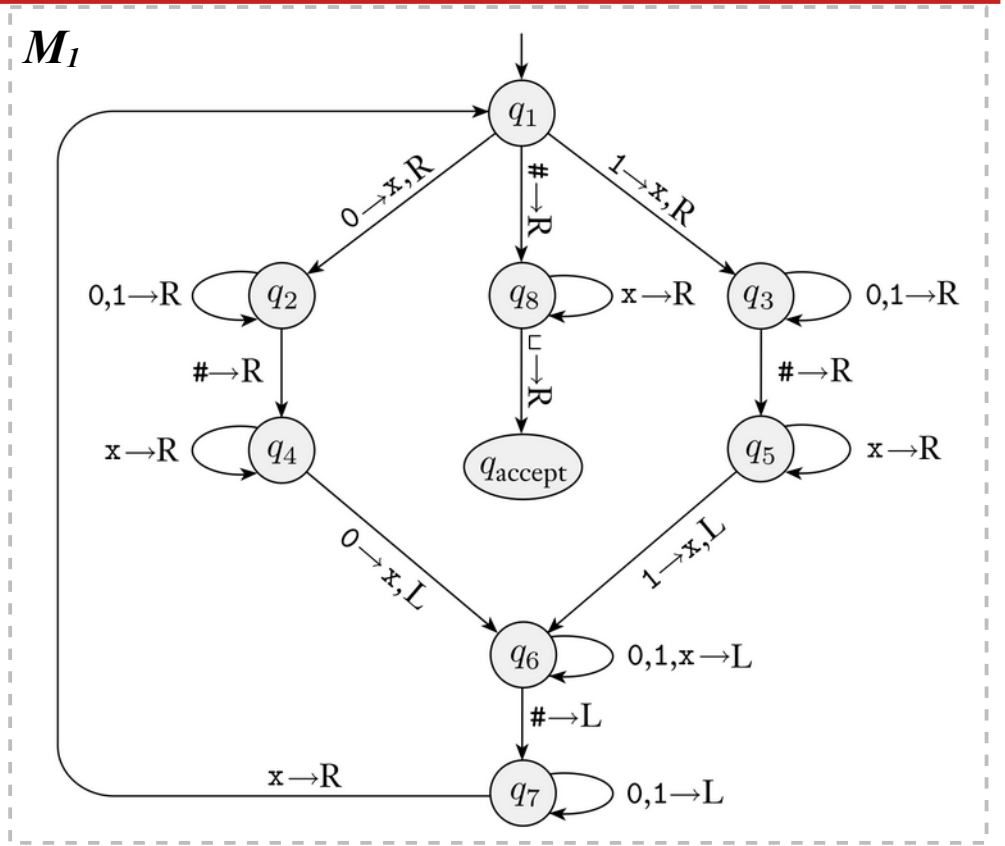
$q_1001\#001\sqcup \Rightarrow xq_201\#001\sqcup$



Turing Machines: Example

Consider $w = 001\#001$

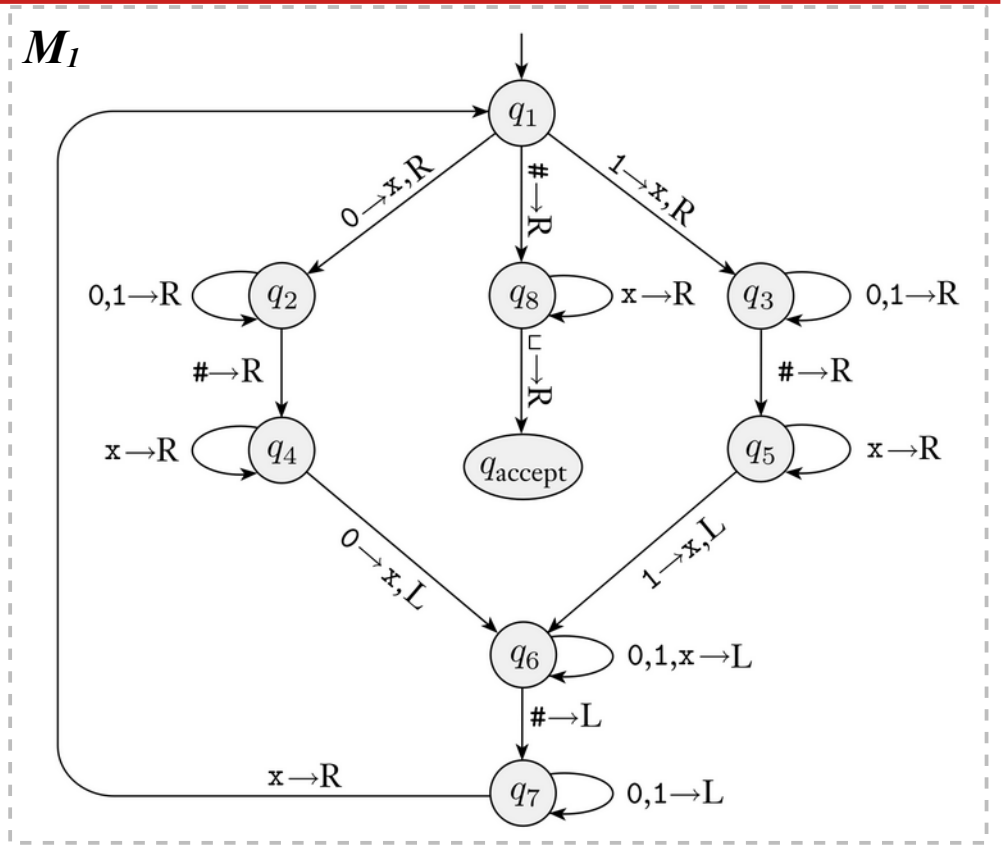
$q_1001\#001_\sqcup \Rightarrow xq_201\#001_\sqcup \Rightarrow x0q_21\#001_\sqcup \Rightarrow$



Turing Machines: Example

Consider $w = 001\#001$

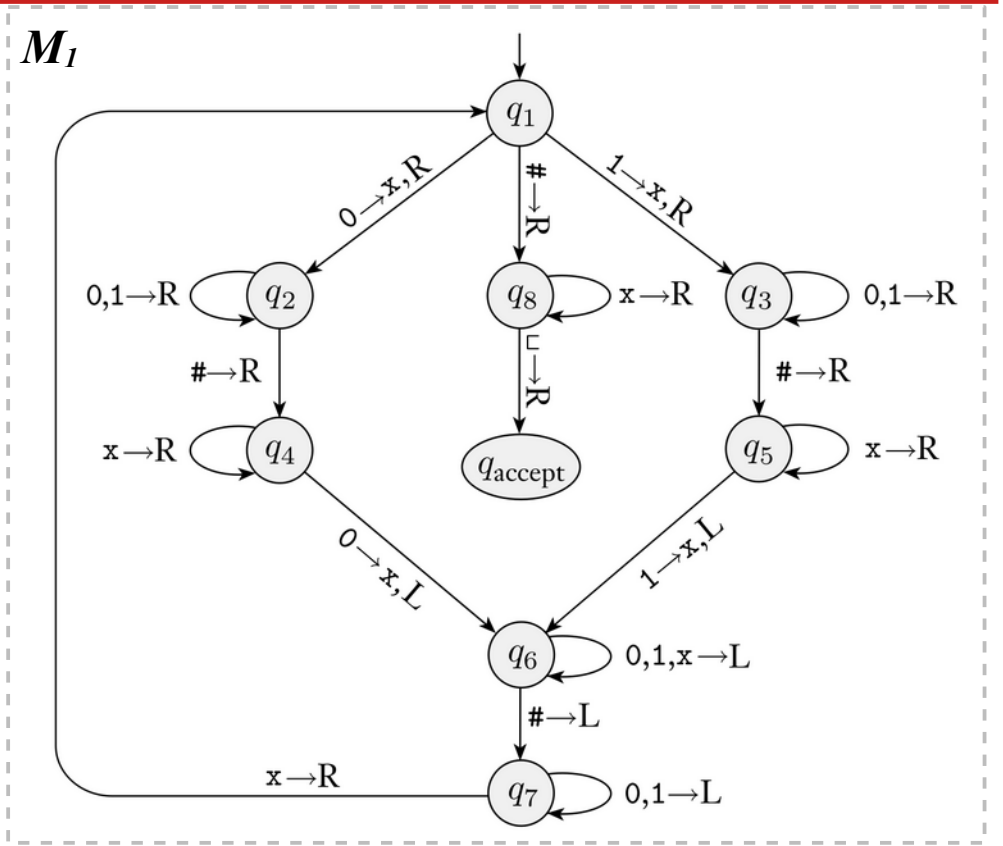
$q_1001\#001\sqcup \Rightarrow xq_201\#001\sqcup \Rightarrow x0q_21\#001\sqcup \Rightarrow$
$x01\#q_4001\sqcup \Rightarrow$



Turing Machines: Example

Consider $w = 001\#001$

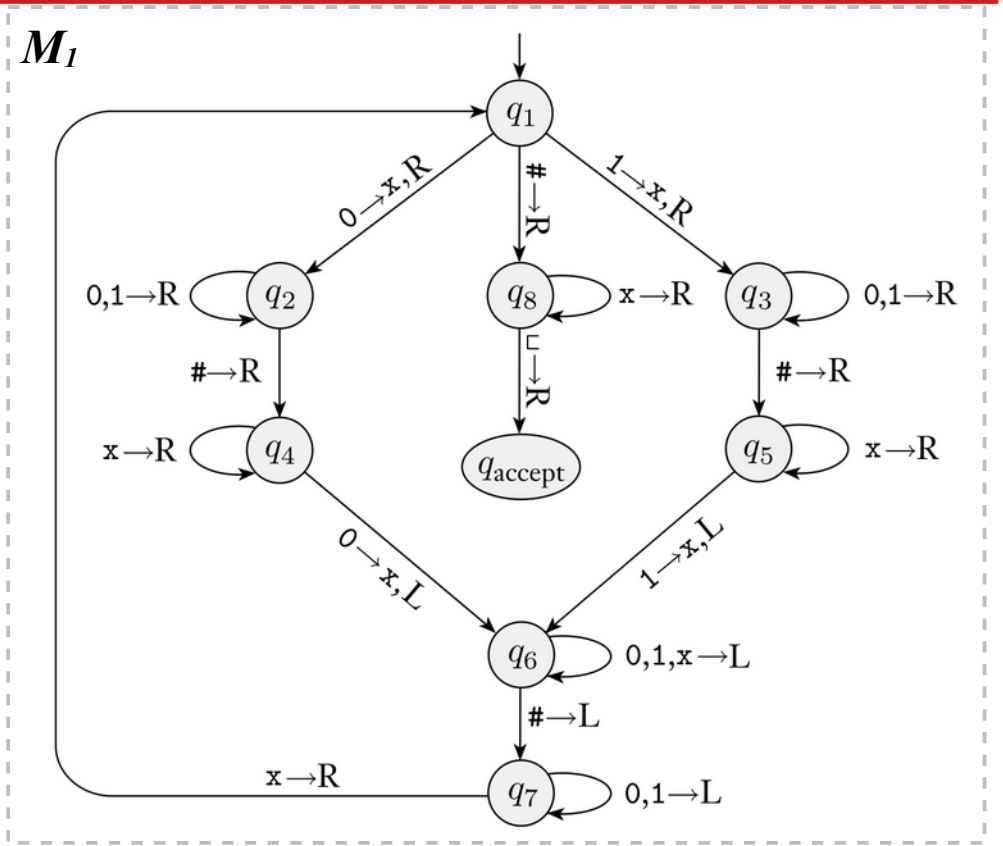
$q_1001\#001\sqcup \Rightarrow xq_201\#001\sqcup \Rightarrow x0q_21\#001\sqcup \Rightarrow$
$x01\#q_4001\sqcup \Rightarrow x01q_6\#x01\sqcup \Rightarrow$



Turing Machines: Example

Consider $w = 001\#001$

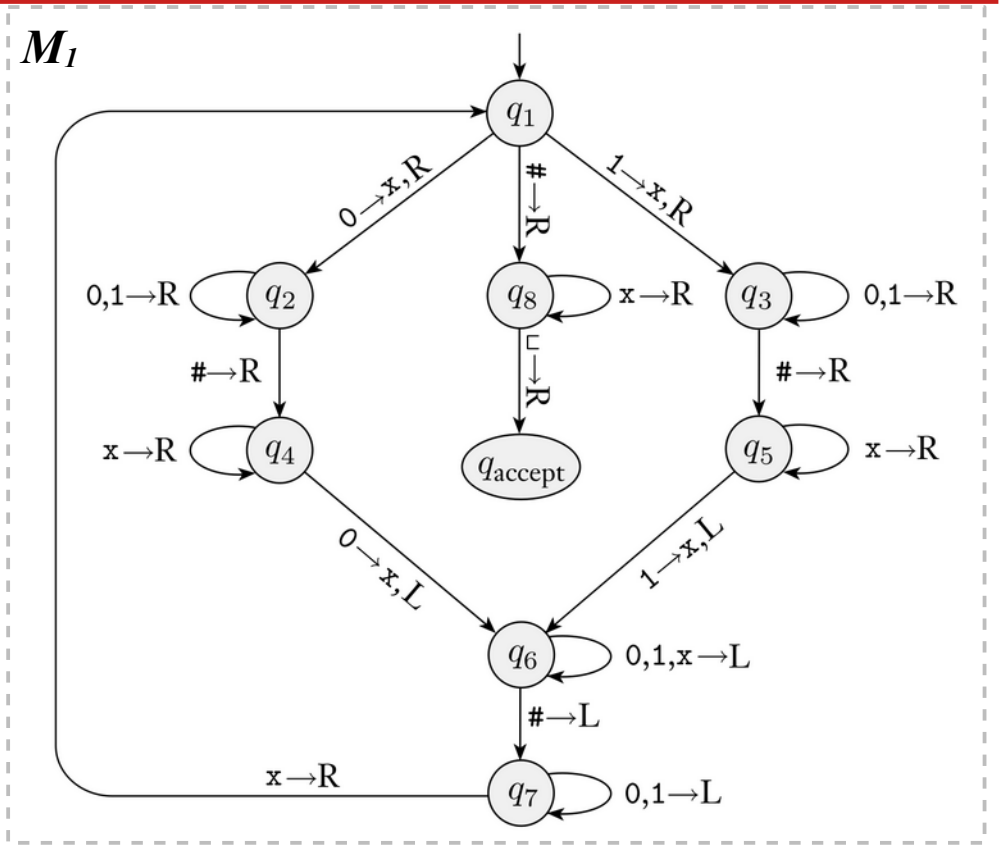
$q_1001\#001_\sqcup \Rightarrow xq_201\#001_\sqcup \Rightarrow x0q_21\#001_\sqcup \Rightarrow$
$x01\#q_4001_\sqcup \Rightarrow x01q_6\#x01_\sqcup \Rightarrow x0q_71\#x01_\sqcup \Rightarrow$



Turing Machines: Example

Consider $w = 001\#001$

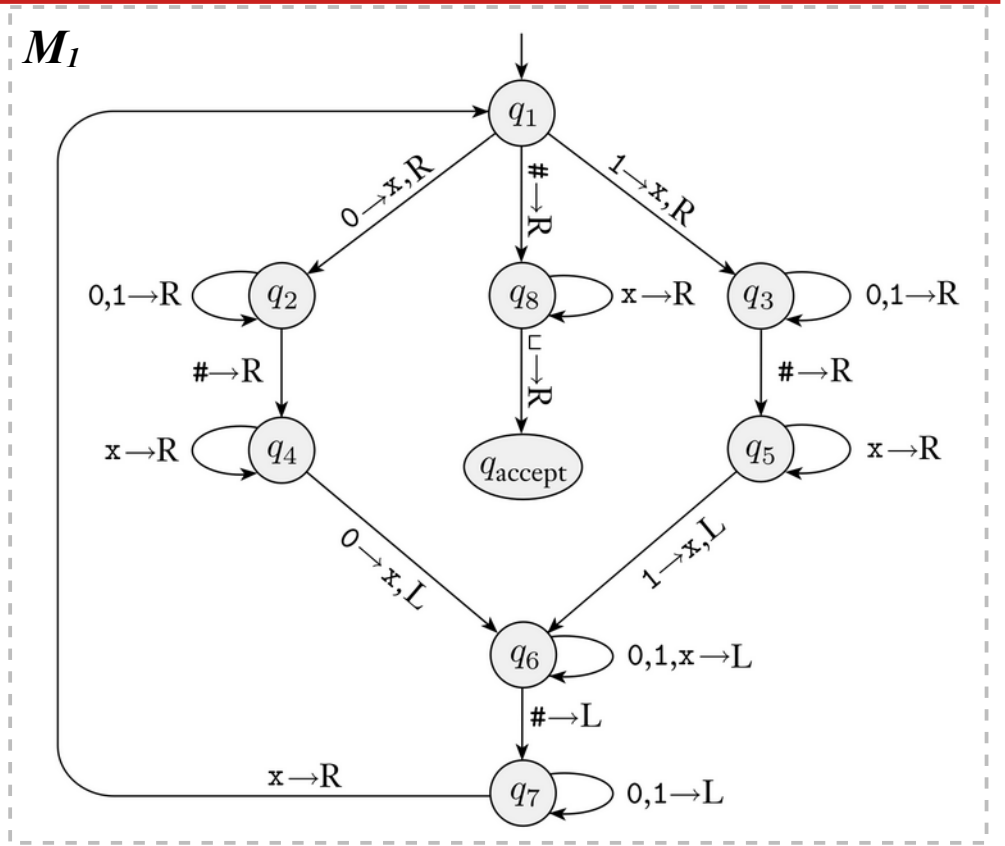
$q_1001\#001_\sqcup \Rightarrow xq_201\#001_\sqcup \Rightarrow x0q_21\#001_\sqcup \Rightarrow$
$x01\#q_4001_\sqcup \Rightarrow x01q_6\#x01_\sqcup \Rightarrow x0q_71\#x01_\sqcup \Rightarrow$
$xq_701\#x01_\sqcup \Rightarrow$



Turing Machines: Example

Consider $w = 001\#001$

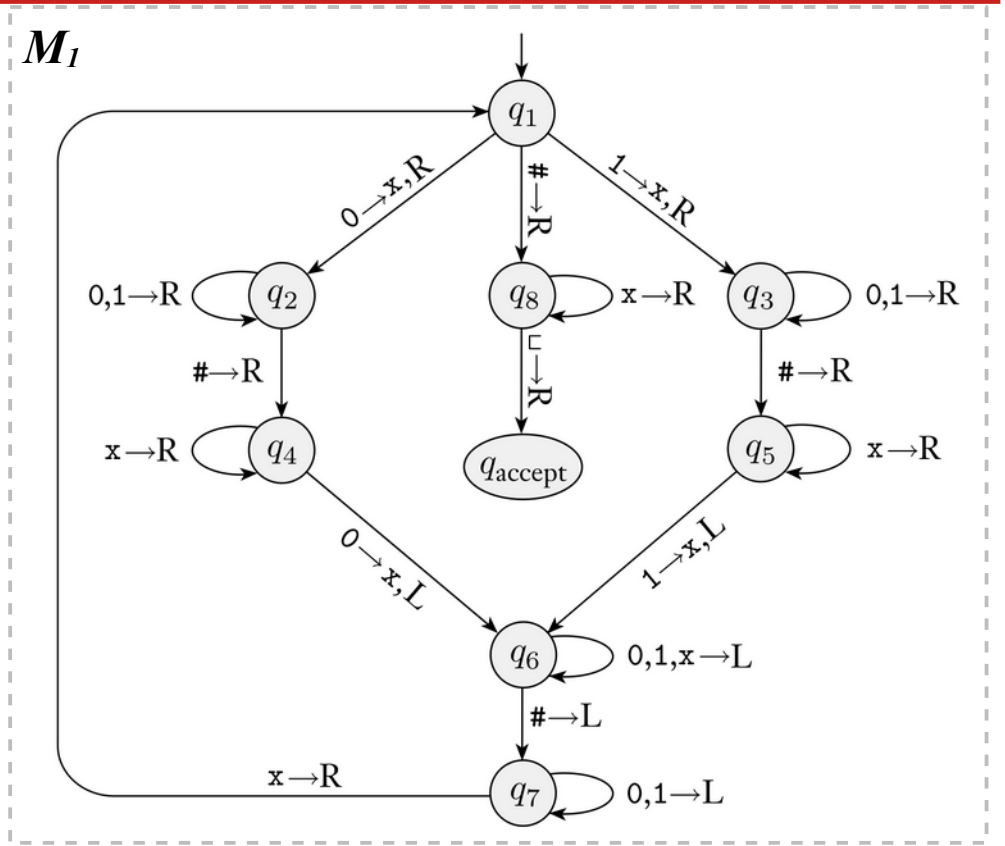
$q_1001\#001_\sqcup \Rightarrow xq_201\#001_\sqcup \Rightarrow x0q_21\#001_\sqcup \Rightarrow$
$x01\#q_4001_\sqcup \Rightarrow x01q_6\#x01_\sqcup \Rightarrow x0q_71\#x01_\sqcup \Rightarrow$
$xq_701\#x01_\sqcup \Rightarrow q_7x01\#x01_\sqcup \Rightarrow$



Turing Machines: Example

Consider $w = 001\#001$

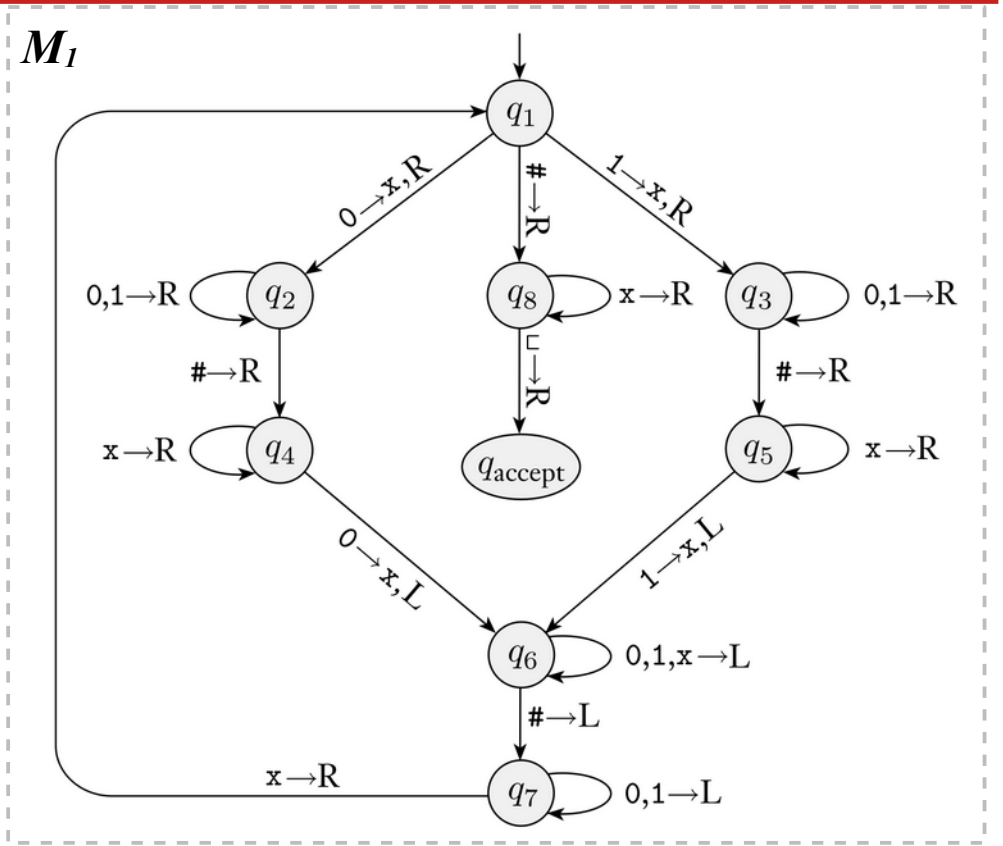
$q_1001\#001_\sqcup \Rightarrow xq_201\#001_\sqcup \Rightarrow x0q_21\#001_\sqcup \Rightarrow$
$x01\#q_4001_\sqcup \Rightarrow x01q_6\#x01_\sqcup \Rightarrow x0q_71\#x01_\sqcup \Rightarrow$
$xq_701\#x01_\sqcup \Rightarrow q_7x01\#x01_\sqcup \Rightarrow xq_101\#x01_\sqcup \Rightarrow$



Turing Machines: Example

Consider $w = 001\#001$

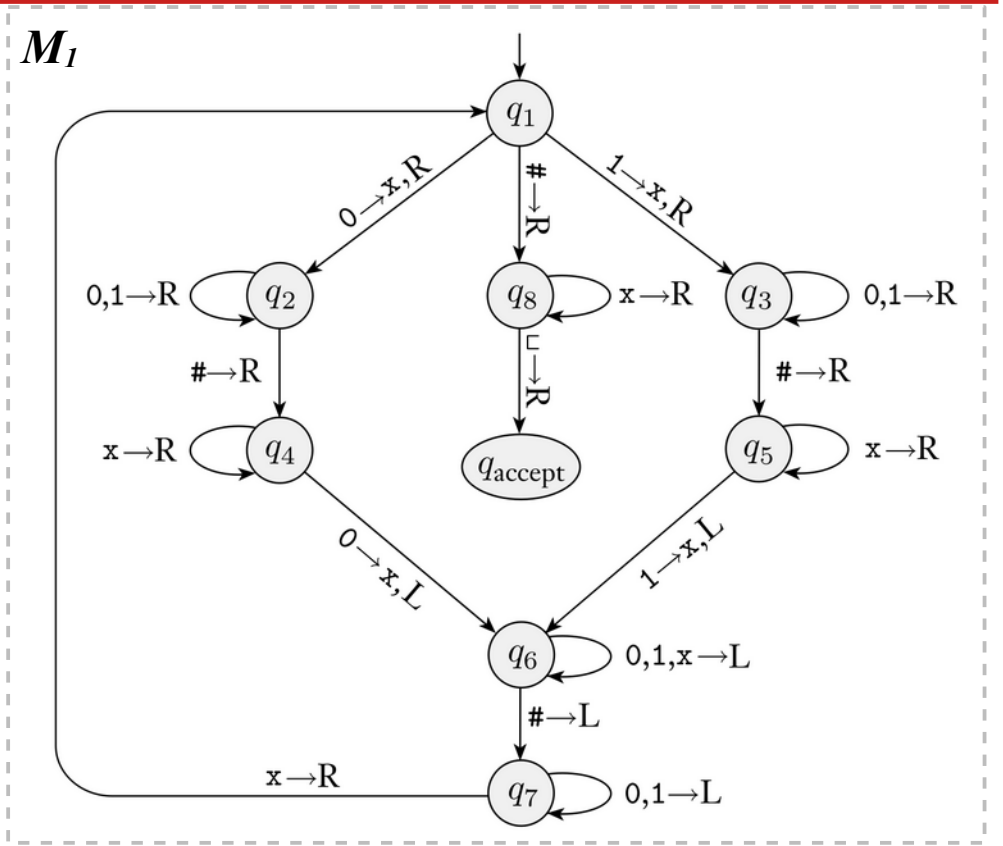
$q_1001\#001_\sqcup \Rightarrow xq_201\#001_\sqcup \Rightarrow x0q_21\#001_\sqcup \Rightarrow$
$x01\#q_4001_\sqcup \Rightarrow x01q_6\#x01_\sqcup \Rightarrow x0q_71\#x01_\sqcup \Rightarrow$
$xq_701\#x01_\sqcup \Rightarrow q_7x01\#x01_\sqcup \Rightarrow xq_101\#x01_\sqcup \Rightarrow$
$xxq_21\#x01_\sqcup \Rightarrow$



Turing Machines: Example

Consider $w = 001\#001$

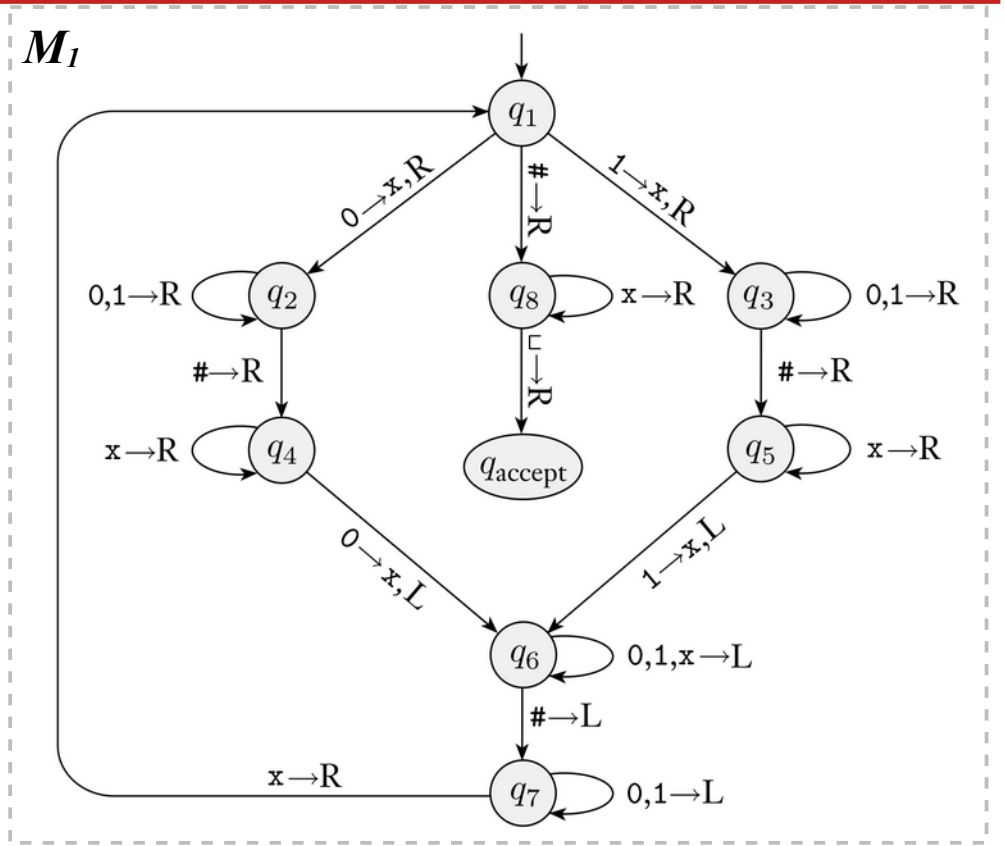
$q_1001\#001_\sqcup \Rightarrow xq_201\#001_\sqcup \Rightarrow x0q_21\#001_\sqcup \Rightarrow$
$x01\#q_4001_\sqcup \Rightarrow x01q_6\#x01_\sqcup \Rightarrow x0q_71\#x01_\sqcup \Rightarrow$
$xq_701\#x01_\sqcup \Rightarrow q_7x01\#x01_\sqcup \Rightarrow xq_101\#x01_\sqcup \Rightarrow$
$xxq_21\#x01_\sqcup \Rightarrow xx1q_2\#x01_\sqcup \Rightarrow$



Turing Machines: Example

Consider $w = 001\#001$

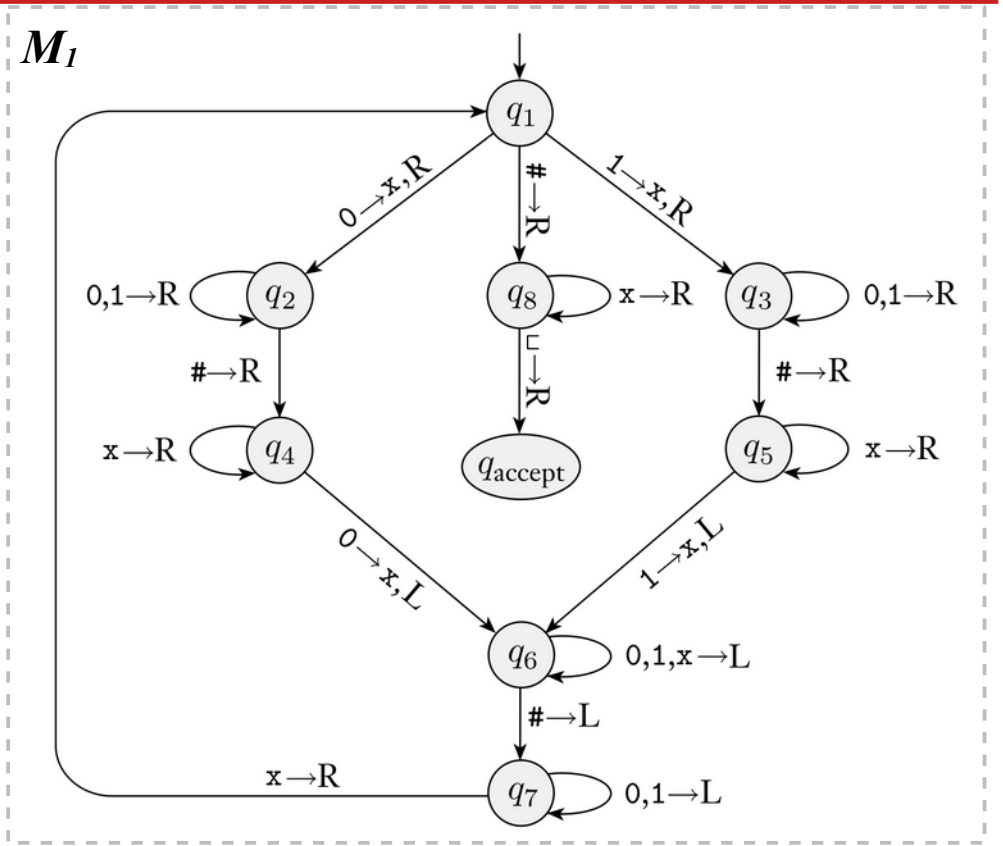
$q_1001\#001_\sqcup \Rightarrow xq_201\#001_\sqcup \Rightarrow x0q_21\#001_\sqcup \Rightarrow$
$x01\#q_4001_\sqcup \Rightarrow x01q_6\#x01_\sqcup \Rightarrow x0q_71\#x01_\sqcup \Rightarrow$
$xq_701\#x01_\sqcup \Rightarrow q_7x01\#x01_\sqcup \Rightarrow xq_101\#x01_\sqcup \Rightarrow$
$xxq_21\#x01_\sqcup \Rightarrow xx1q_2\#x01_\sqcup \Rightarrow xx1\#q_4x01_\sqcup \Rightarrow$



Turing Machines: Example

Consider $w = 001\#001$

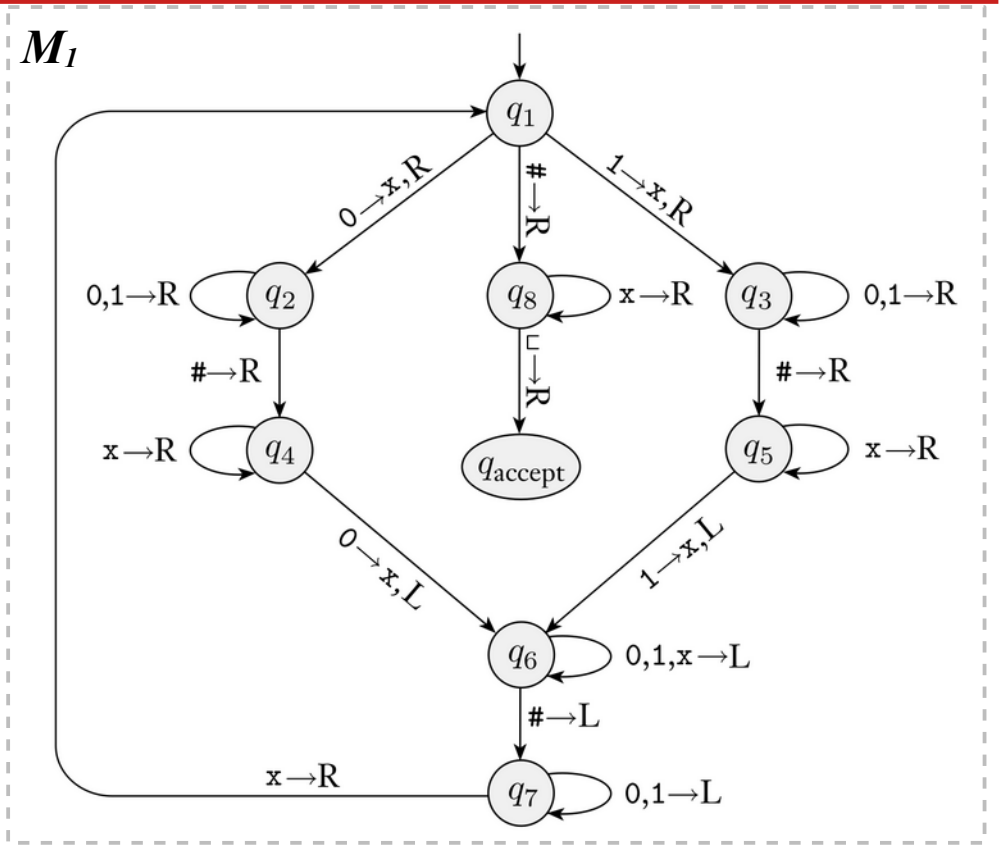
$q_1001\#001_\sqcup \Rightarrow xq_201\#001_\sqcup \Rightarrow x0q_21\#001_\sqcup \Rightarrow$
$x01\#q_4001_\sqcup \Rightarrow x01q_6\#x01_\sqcup \Rightarrow x0q_71\#x01_\sqcup \Rightarrow$
$xq_701\#x01_\sqcup \Rightarrow q_7x01\#x01_\sqcup \Rightarrow xq_101\#x01_\sqcup \Rightarrow$
$xxq_21\#x01_\sqcup \Rightarrow xx1q_2\#x01_\sqcup \Rightarrow xx1\#q_4x01_\sqcup \Rightarrow$
$xx1\#xq_401_\sqcup \Rightarrow$



Turing Machines: Example

Consider $w = 001\#001$

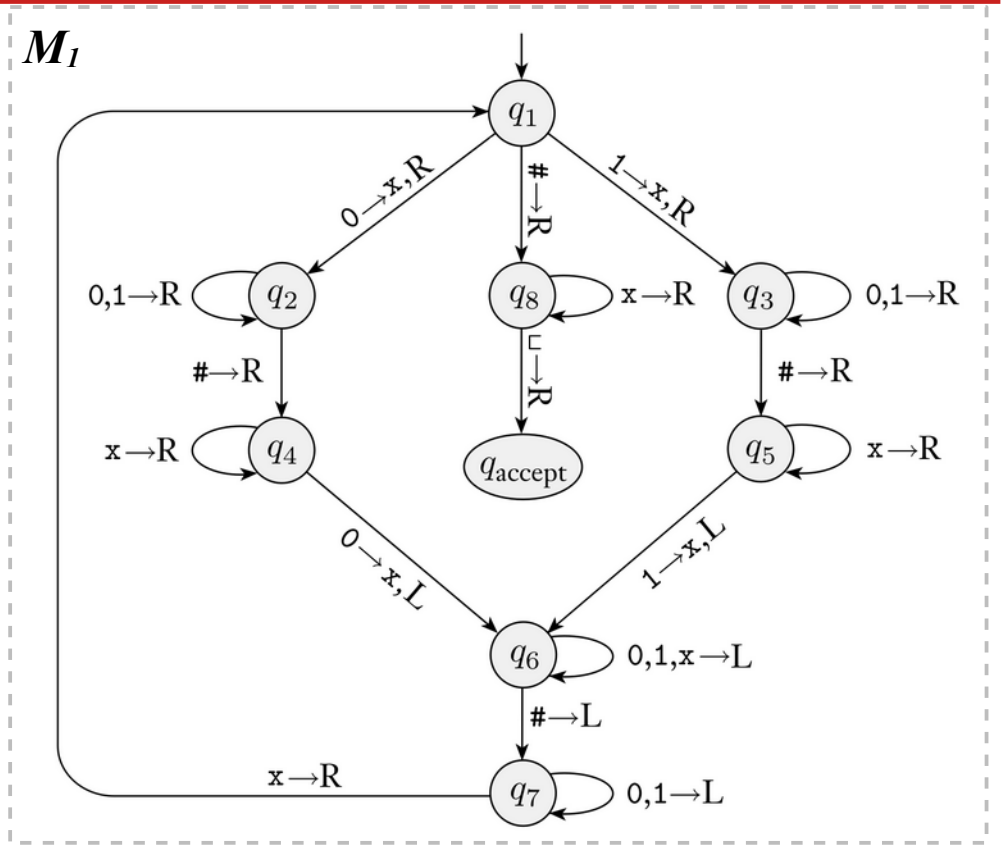
$q_1001\#001_\sqcup \Rightarrow xq_201\#001_\sqcup \Rightarrow x0q_21\#001_\sqcup \Rightarrow$
$x01\#q_4001_\sqcup \Rightarrow x01q_6\#x01_\sqcup \Rightarrow x0q_71\#x01_\sqcup \Rightarrow$
$xq_701\#x01_\sqcup \Rightarrow q_7x01\#x01_\sqcup \Rightarrow xq_101\#x01_\sqcup \Rightarrow$
$xxq_21\#x01_\sqcup \Rightarrow xx1q_2\#x01_\sqcup \Rightarrow xx1\#q_4x01_\sqcup \Rightarrow$
$xx1\#xq_401_\sqcup \Rightarrow xx1\#q_6xx1_\sqcup \Rightarrow$



Turing Machines: Example

Consider $w = 001\#001$

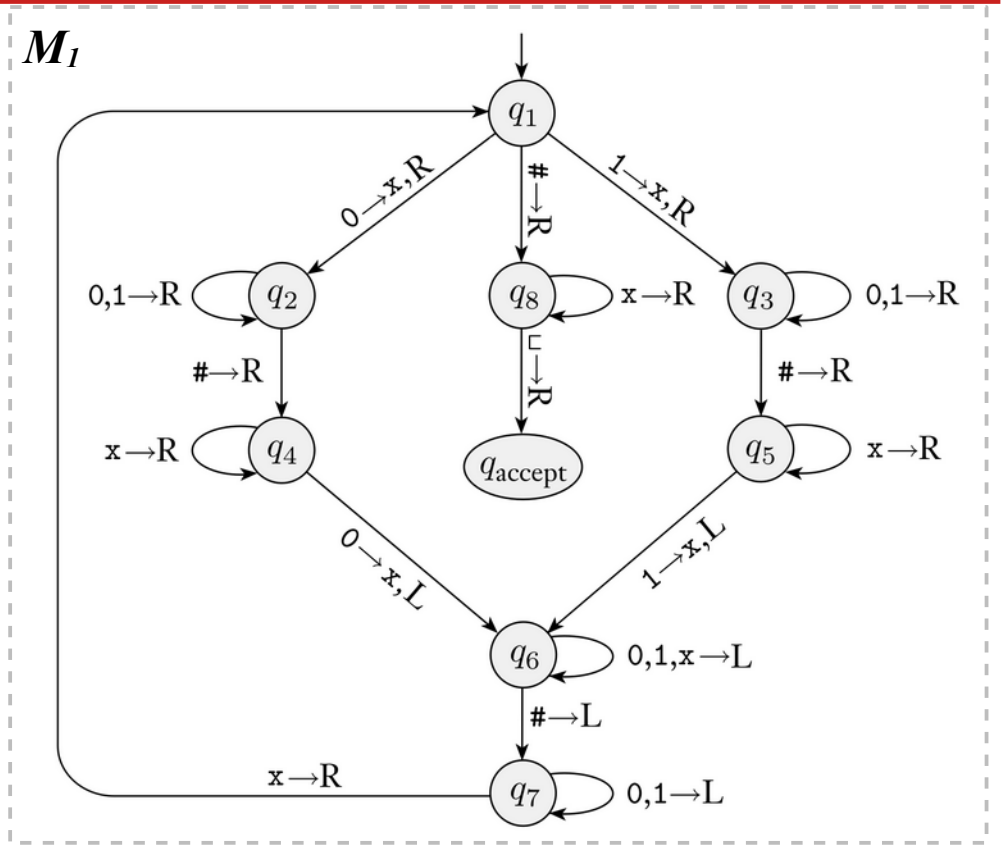
$q_1001\#001_\sqcup \Rightarrow xq_201\#001_\sqcup \Rightarrow x0q_21\#001_\sqcup \Rightarrow$
$x01\#q_4001_\sqcup \Rightarrow x01q_6\#x01_\sqcup \Rightarrow x0q_71\#x01_\sqcup \Rightarrow$
$xq_701\#x01_\sqcup \Rightarrow q_7x01\#x01_\sqcup \Rightarrow xq_101\#x01_\sqcup \Rightarrow$
$xxq_21\#x01_\sqcup \Rightarrow xx1q_2\#x01_\sqcup \Rightarrow xx1\#q_4x01_\sqcup \Rightarrow$
$xx1\#xq_401_\sqcup \Rightarrow xx1\#q_6xx1_\sqcup \Rightarrow xx1q_6\#xx1_\sqcup \Rightarrow$



Turing Machines: Example

Consider $w = 001\#001$

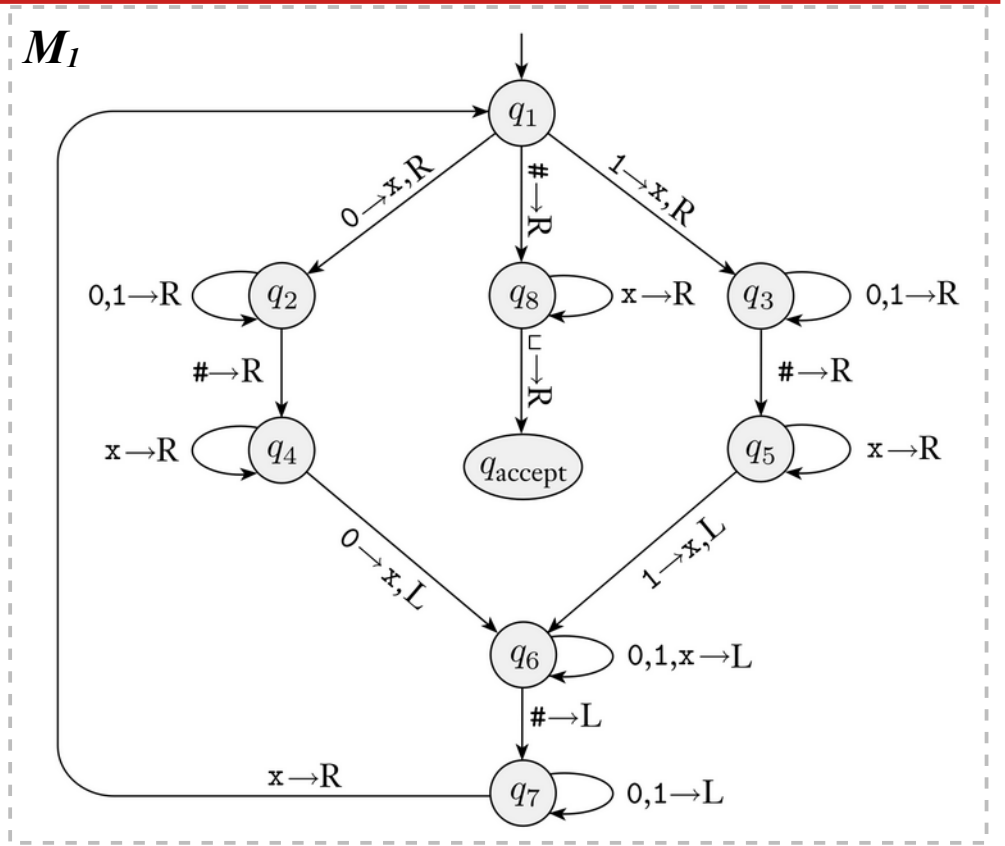
$q_1001\#001_\sqcup \Rightarrow xq_201\#001_\sqcup \Rightarrow x0q_21\#001_\sqcup \Rightarrow$
$x01\#q_4001_\sqcup \Rightarrow x01q_6\#x01_\sqcup \Rightarrow x0q_71\#x01_\sqcup \Rightarrow$
$xq_701\#x01_\sqcup \Rightarrow q_7x01\#x01_\sqcup \Rightarrow xq_101\#x01_\sqcup \Rightarrow$
$xxq_21\#x01_\sqcup \Rightarrow xx1q_2\#x01_\sqcup \Rightarrow xx1\#q_4x01_\sqcup \Rightarrow$
$xx1\#xq_401_\sqcup \Rightarrow xx1\#q_6xx1_\sqcup \Rightarrow xx1q_6\#xx1_\sqcup \Rightarrow$
$xxq_71\#xx1_\sqcup \Rightarrow$



Turing Machines: Example

Consider $w = 001\#001$

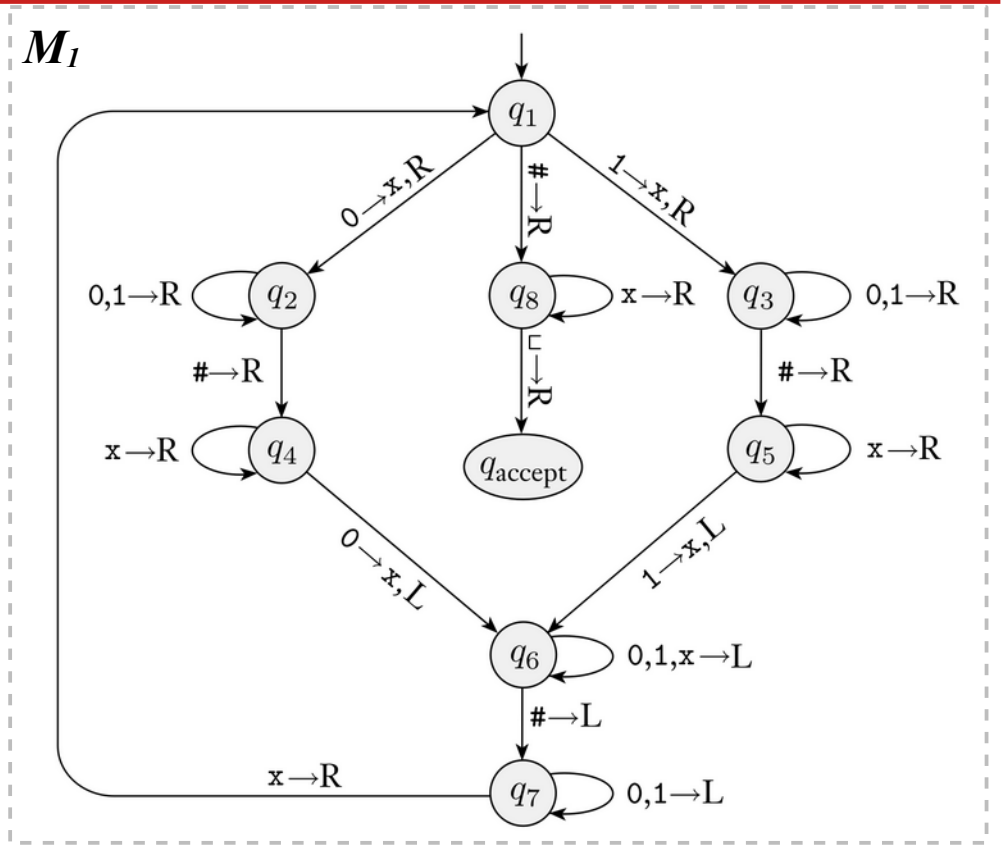
$q_1001\#001_\sqcup \Rightarrow xq_201\#001_\sqcup \Rightarrow x0q_21\#001_\sqcup \Rightarrow$
$x01\#q_4001_\sqcup \Rightarrow x01q_6\#x01_\sqcup \Rightarrow x0q_71\#x01_\sqcup \Rightarrow$
$xq_701\#x01_\sqcup \Rightarrow q_7x01\#x01_\sqcup \Rightarrow xq_101\#x01_\sqcup \Rightarrow$
$xxq_21\#x01_\sqcup \Rightarrow xx1q_2\#x01_\sqcup \Rightarrow xx1\#q_4x01_\sqcup \Rightarrow$
$xx1\#xq_401_\sqcup \Rightarrow xx1\#q_6xx1_\sqcup \Rightarrow xx1q_6\#xx1_\sqcup \Rightarrow$
$xxq_71\#xx1_\sqcup \Rightarrow xq_7x1\#xx1_\sqcup \Rightarrow$



Turing Machines: Example

Consider $w = 001\#001$

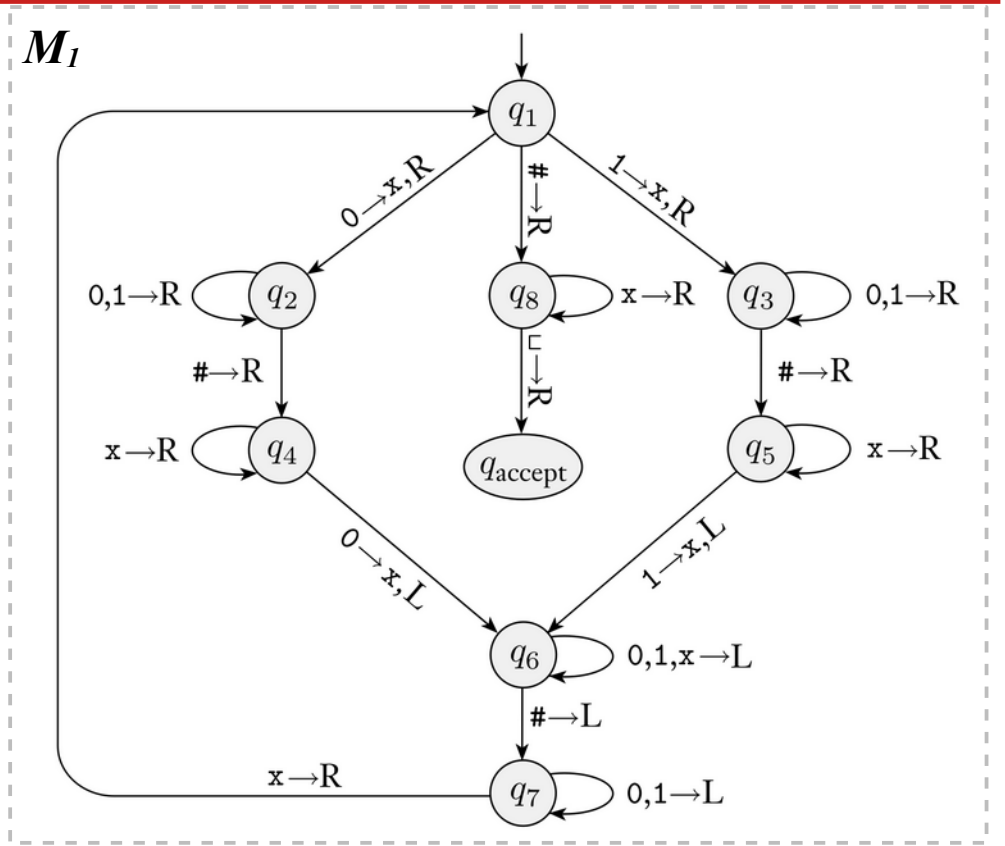
$q_1001\#001_\sqcup \Rightarrow xq_201\#001_\sqcup \Rightarrow x0q_21\#001_\sqcup \Rightarrow$
$x01\#q_4001_\sqcup \Rightarrow x01q_6\#x01_\sqcup \Rightarrow x0q_71\#x01_\sqcup \Rightarrow$
$xq_701\#x01_\sqcup \Rightarrow q_7x01\#x01_\sqcup \Rightarrow xq_101\#x01_\sqcup \Rightarrow$
$xxq_21\#x01_\sqcup \Rightarrow xx1q_2\#x01_\sqcup \Rightarrow xx1\#q_4x01_\sqcup \Rightarrow$
$xx1\#xq_401_\sqcup \Rightarrow xx1\#q_6xx1_\sqcup \Rightarrow xx1q_6\#xx1_\sqcup \Rightarrow$
$xxq_71\#xx1_\sqcup \Rightarrow xq_7x1\#xx1_\sqcup \Rightarrow xxq_11\#xx1_\sqcup \Rightarrow$



Turing Machines: Example

Consider $w = 001\#001$

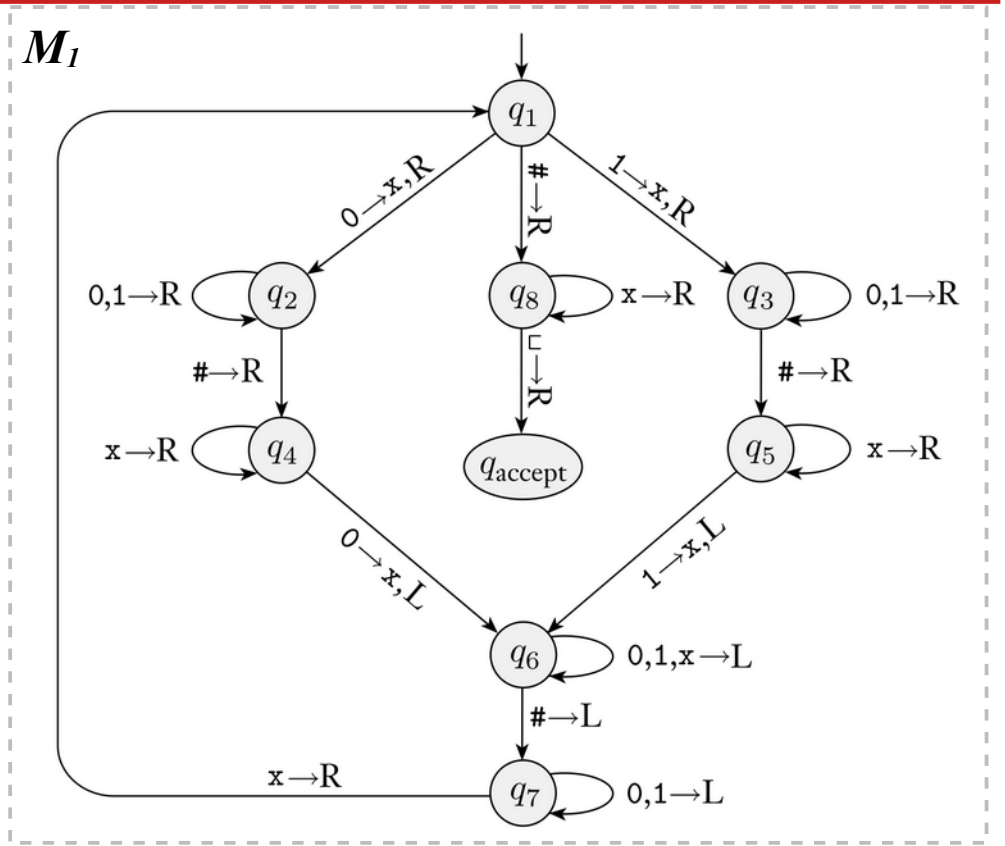
$q_1001\#001_\sqcup \Rightarrow xq_201\#001_\sqcup \Rightarrow x0q_21\#001_\sqcup \Rightarrow$
$x01\#q_4001_\sqcup \Rightarrow x01q_6\#x01_\sqcup \Rightarrow x0q_71\#x01_\sqcup \Rightarrow$
$xq_701\#x01_\sqcup \Rightarrow q_7x01\#x01_\sqcup \Rightarrow xq_101\#x01_\sqcup \Rightarrow$
$xxq_21\#x01_\sqcup \Rightarrow xx1q_2\#x01_\sqcup \Rightarrow xx1\#q_4x01_\sqcup \Rightarrow$
$xx1\#xq_401_\sqcup \Rightarrow xx1\#q_6xx1_\sqcup \Rightarrow xx1q_6\#xx1_\sqcup \Rightarrow$
$xxq_71\#xx1_\sqcup \Rightarrow xq_7x1\#xx1_\sqcup \Rightarrow xxq_11\#xx1_\sqcup \Rightarrow$
$xxxq_3\#xx1_\sqcup \Rightarrow$



Turing Machines: Example

Consider $w = 001\#001$

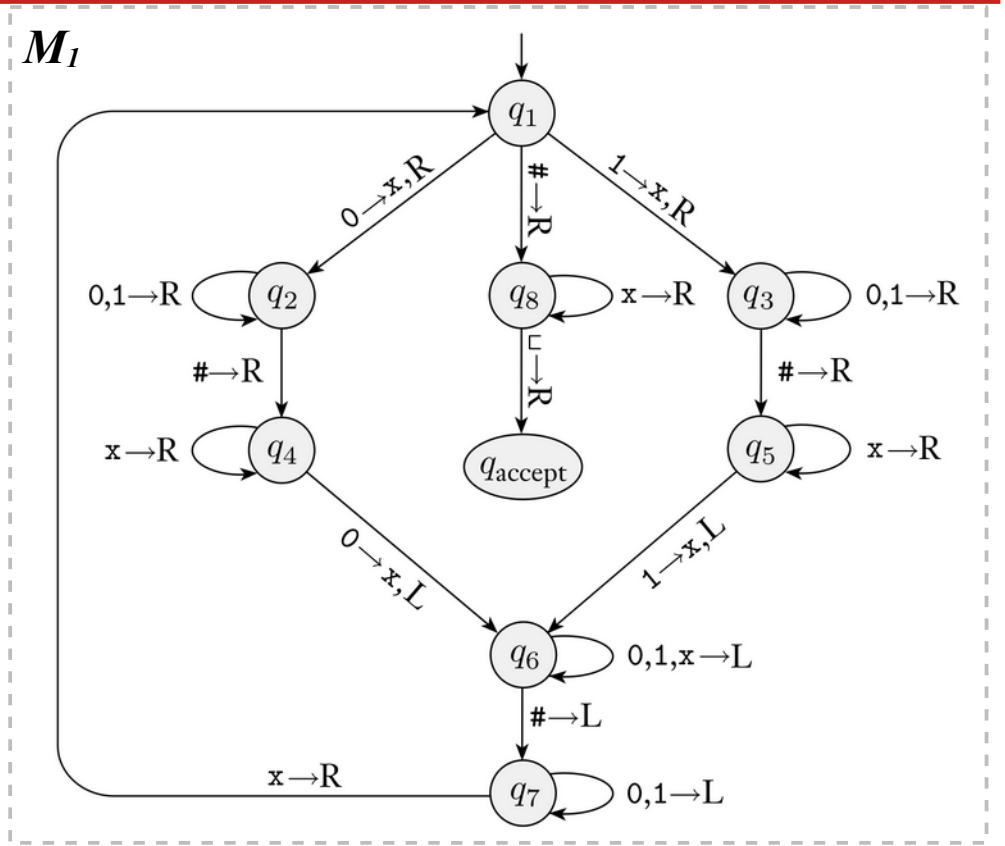
$q_1001\#001_\sqcup \Rightarrow xq_201\#001_\sqcup \Rightarrow x0q_21\#001_\sqcup \Rightarrow$
$x01\#q_4001_\sqcup \Rightarrow x01q_6\#x01_\sqcup \Rightarrow x0q_71\#x01_\sqcup \Rightarrow$
$xq_701\#x01_\sqcup \Rightarrow q_7x01\#x01_\sqcup \Rightarrow xq_101\#x01_\sqcup \Rightarrow$
$xxq_21\#x01_\sqcup \Rightarrow xx1q_2\#x01_\sqcup \Rightarrow xx1\#q_4x01_\sqcup \Rightarrow$
$xx1\#xq_401_\sqcup \Rightarrow xx1\#q_6xx1_\sqcup \Rightarrow xx1q_6\#xx1_\sqcup \Rightarrow$
$xxq_71\#xx1_\sqcup \Rightarrow xq_7x1\#xx1_\sqcup \Rightarrow xxq_11\#xx1_\sqcup \Rightarrow$
$xxxq_3\#xx1_\sqcup \Rightarrow xxx\#q_5xx1_\sqcup \Rightarrow$



Turing Machines: Example

Consider $w = 001\#001$

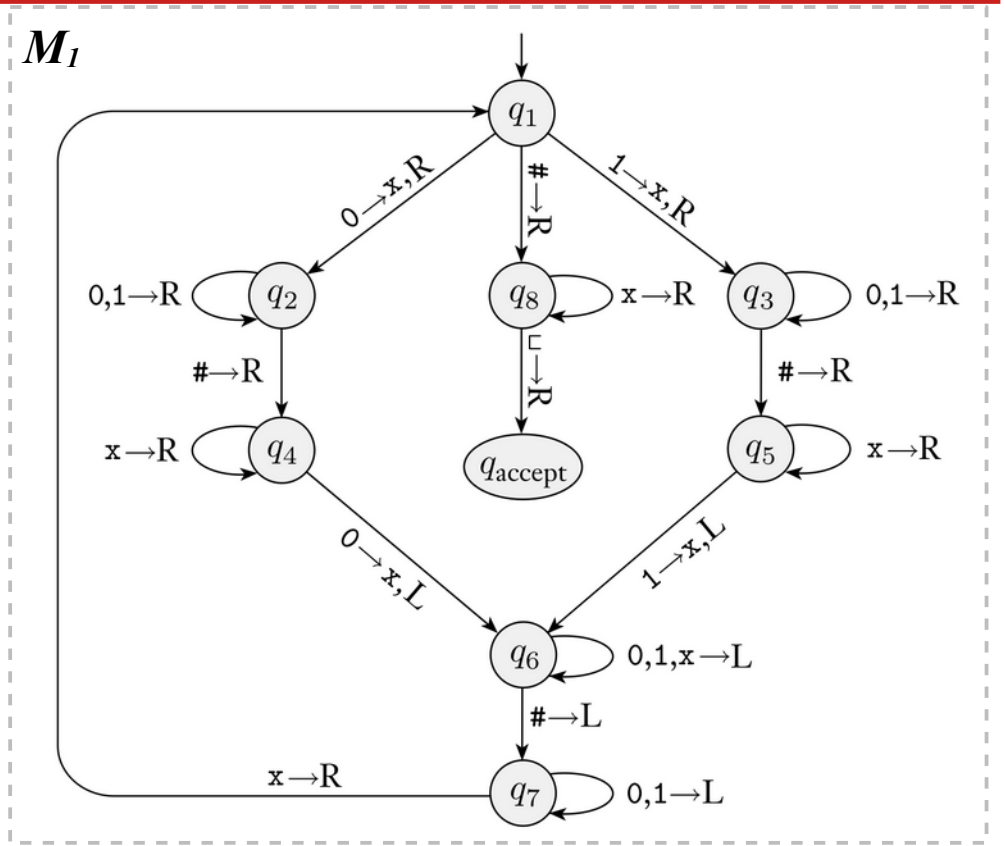
$q_1001\#001_\sqcup \Rightarrow xq_201\#001_\sqcup \Rightarrow x0q_21\#001_\sqcup \Rightarrow$
$x01\#q_4001_\sqcup \Rightarrow x01q_6\#x01_\sqcup \Rightarrow x0q_71\#x01_\sqcup \Rightarrow$
$xq_701\#x01_\sqcup \Rightarrow q_7x01\#x01_\sqcup \Rightarrow xq_101\#x01_\sqcup \Rightarrow$
$xxq_21\#x01_\sqcup \Rightarrow xx1q_2\#x01_\sqcup \Rightarrow xx1\#q_4x01_\sqcup \Rightarrow$
$xx1\#xq_401_\sqcup \Rightarrow xx1\#q_6xx1_\sqcup \Rightarrow xx1q_6\#xx1_\sqcup \Rightarrow$
$xxq_71\#xx1_\sqcup \Rightarrow xq_7x1\#xx1_\sqcup \Rightarrow xxq_11\#xx1_\sqcup \Rightarrow$
$xxxq_3\#xx1_\sqcup \Rightarrow xxx\#q_5xx1_\sqcup \Rightarrow xxx\#xq_5x1_\sqcup \Rightarrow$



Turing Machines: Example

Consider $w = 001\#001$

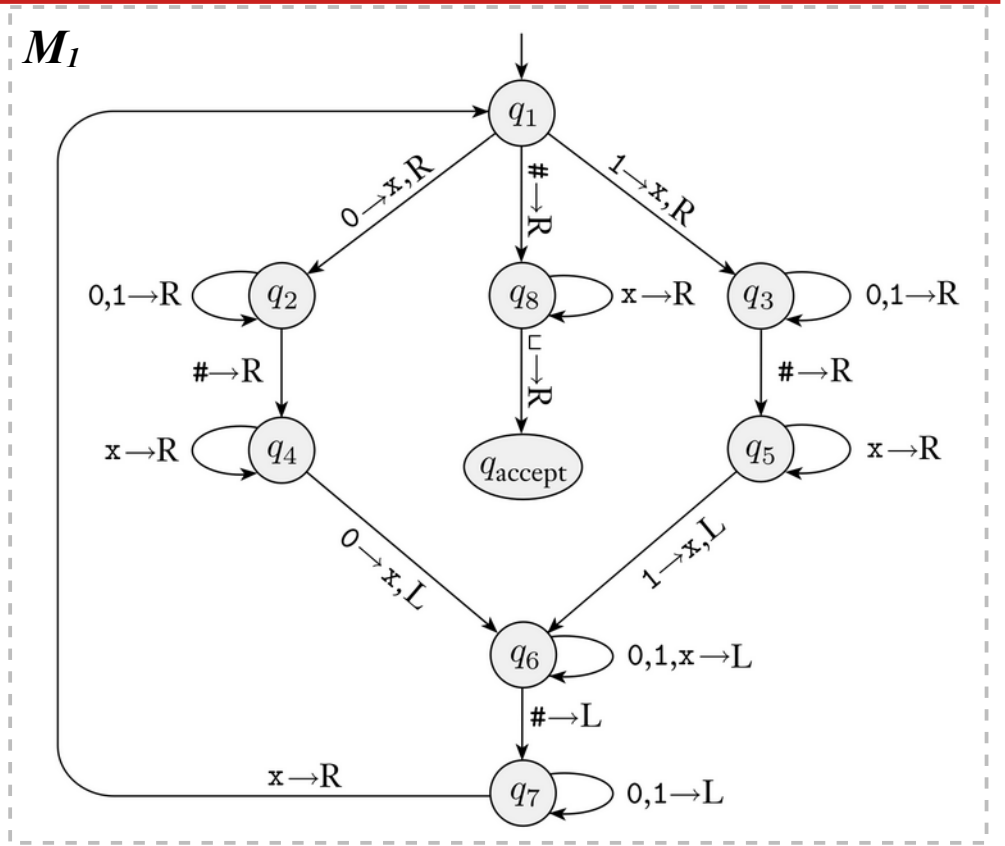
$q_1001\#001_\sqcup \Rightarrow xq_201\#001_\sqcup \Rightarrow x0q_21\#001_\sqcup \Rightarrow$
$x01\#q_4001_\sqcup \Rightarrow x01q_6\#x01_\sqcup \Rightarrow x0q_71\#x01_\sqcup \Rightarrow$
$xq_701\#x01_\sqcup \Rightarrow q_7x01\#x01_\sqcup \Rightarrow xq_101\#x01_\sqcup \Rightarrow$
$xxq_21\#x01_\sqcup \Rightarrow xx1q_2\#x01_\sqcup \Rightarrow xx1\#q_4x01_\sqcup \Rightarrow$
$xx1\#xq_401_\sqcup \Rightarrow xx1\#q_6xx1_\sqcup \Rightarrow xx1q_6\#xx1_\sqcup \Rightarrow$
$xxq_71\#xx1_\sqcup \Rightarrow xq_7x1\#xx1_\sqcup \Rightarrow xxq_11\#xx1_\sqcup \Rightarrow$
$xxxq_3\#xx1_\sqcup \Rightarrow xxx\#q_5xx1_\sqcup \Rightarrow xxx\#xq_5x1_\sqcup \Rightarrow$
$xxx\#xxq_51_\sqcup \Rightarrow$



Turing Machines: Example

Consider $w = 001\#001$

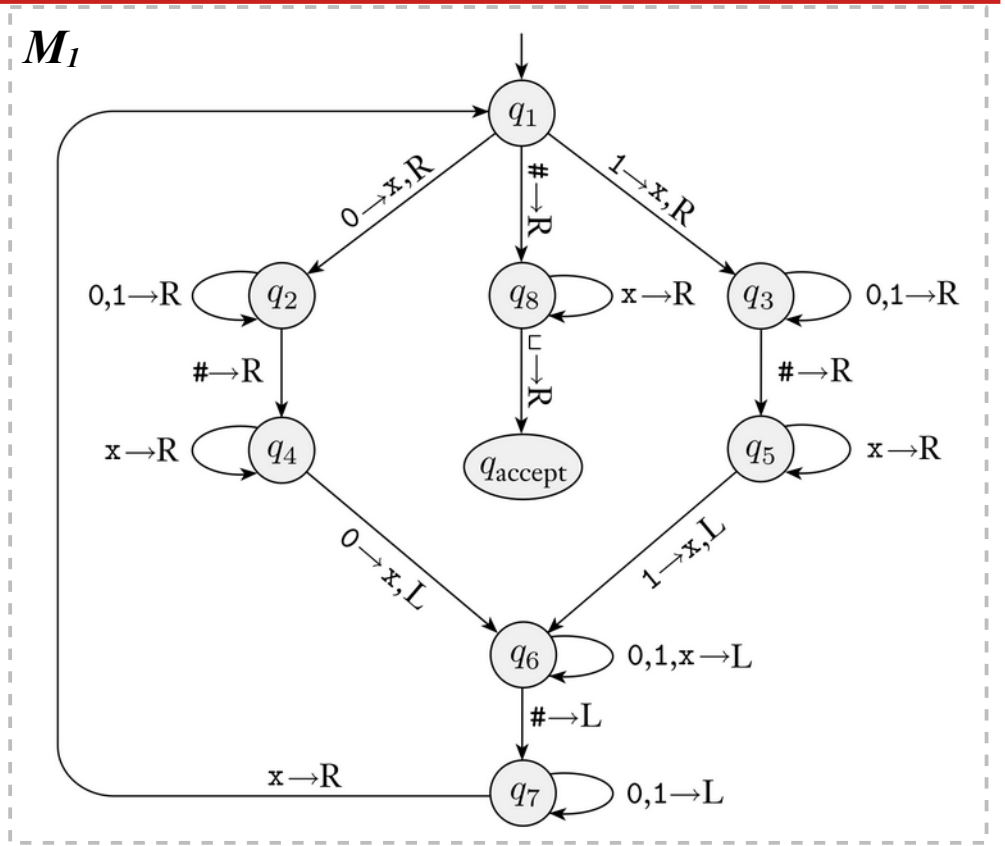
$q_1001\#001_\sqcup \Rightarrow xq_201\#001_\sqcup \Rightarrow x0q_21\#001_\sqcup \Rightarrow$
$x01\#q_4001_\sqcup \Rightarrow x01q_6\#x01_\sqcup \Rightarrow x0q_71\#x01_\sqcup \Rightarrow$
$xq_701\#x01_\sqcup \Rightarrow q_7x01\#x01_\sqcup \Rightarrow xq_101\#x01_\sqcup \Rightarrow$
$xxq_21\#x01_\sqcup \Rightarrow xx1q_2\#x01_\sqcup \Rightarrow xx1\#q_4x01_\sqcup \Rightarrow$
$xx1\#xq_401_\sqcup \Rightarrow xx1\#q_6xx1_\sqcup \Rightarrow xx1q_6\#xx1_\sqcup \Rightarrow$
$xxq_71\#xx1_\sqcup \Rightarrow xq_7x1\#xx1_\sqcup \Rightarrow xxq_11\#xx1_\sqcup \Rightarrow$
$xxxq_3\#xx1_\sqcup \Rightarrow xxx\#q_5xx1_\sqcup \Rightarrow xxx\#xq_5x1_\sqcup \Rightarrow$
$xxx\#xxq_51_\sqcup \Rightarrow xxx\#xq_6xx \Rightarrow$



Turing Machines: Example

Consider $w = 001\#001$

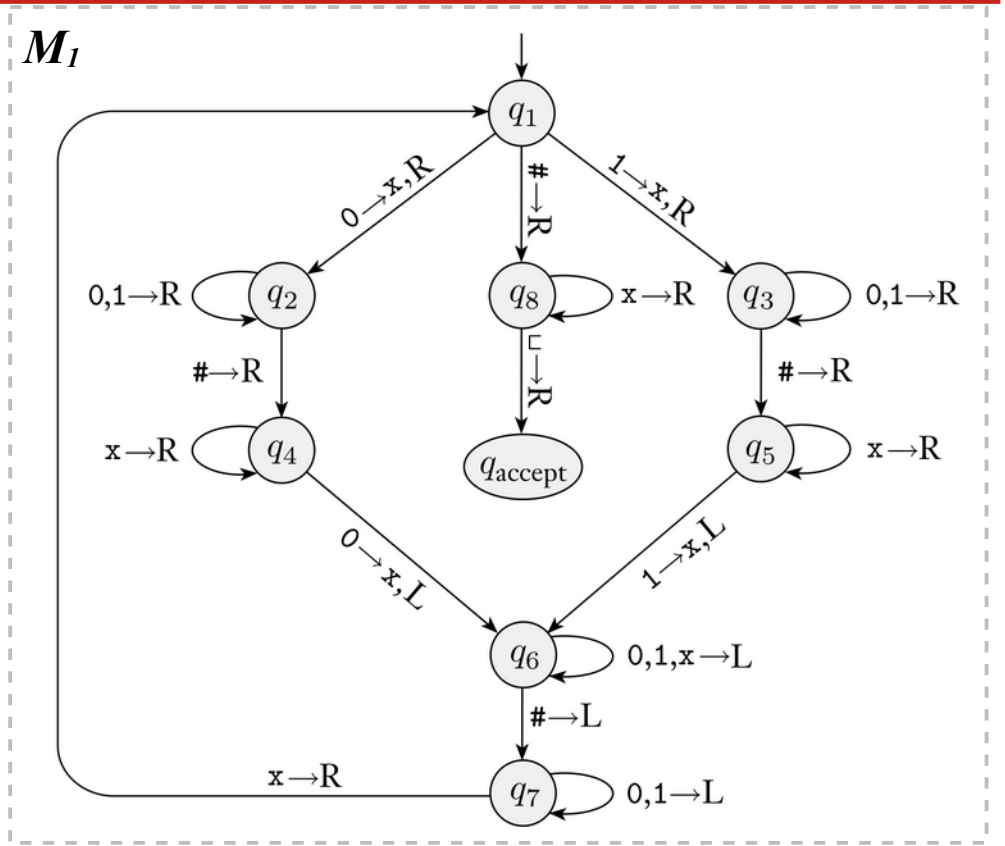
$q_1001\#001_\sqcup \Rightarrow xq_201\#001_\sqcup \Rightarrow x0q_21\#001_\sqcup \Rightarrow$
$x01\#q_4001_\sqcup \Rightarrow x01q_6\#x01_\sqcup \Rightarrow x0q_71\#x01_\sqcup \Rightarrow$
$xq_701\#x01_\sqcup \Rightarrow q_7x01\#x01_\sqcup \Rightarrow xq_101\#x01_\sqcup \Rightarrow$
$xxq_21\#x01_\sqcup \Rightarrow xx1q_2\#x01_\sqcup \Rightarrow xx1\#q_4x01_\sqcup \Rightarrow$
$xx1\#xq_401_\sqcup \Rightarrow xx1\#q_6xx1_\sqcup \Rightarrow xx1q_6\#xx1_\sqcup \Rightarrow$
$xxq_71\#xx1_\sqcup \Rightarrow xq_7x1\#xx1_\sqcup \Rightarrow xxq_11\#xx1_\sqcup \Rightarrow$
$xxxq_3\#xx1_\sqcup \Rightarrow xxx\#q_5xx1_\sqcup \Rightarrow xxx\#xq_5x1_\sqcup \Rightarrow$
$xxx\#xxq_51_\sqcup \Rightarrow xxx\#xq_6xx \Rightarrow xxx\#q_6xxx \Rightarrow$



Turing Machines: Example

Consider $w = 001\#001$

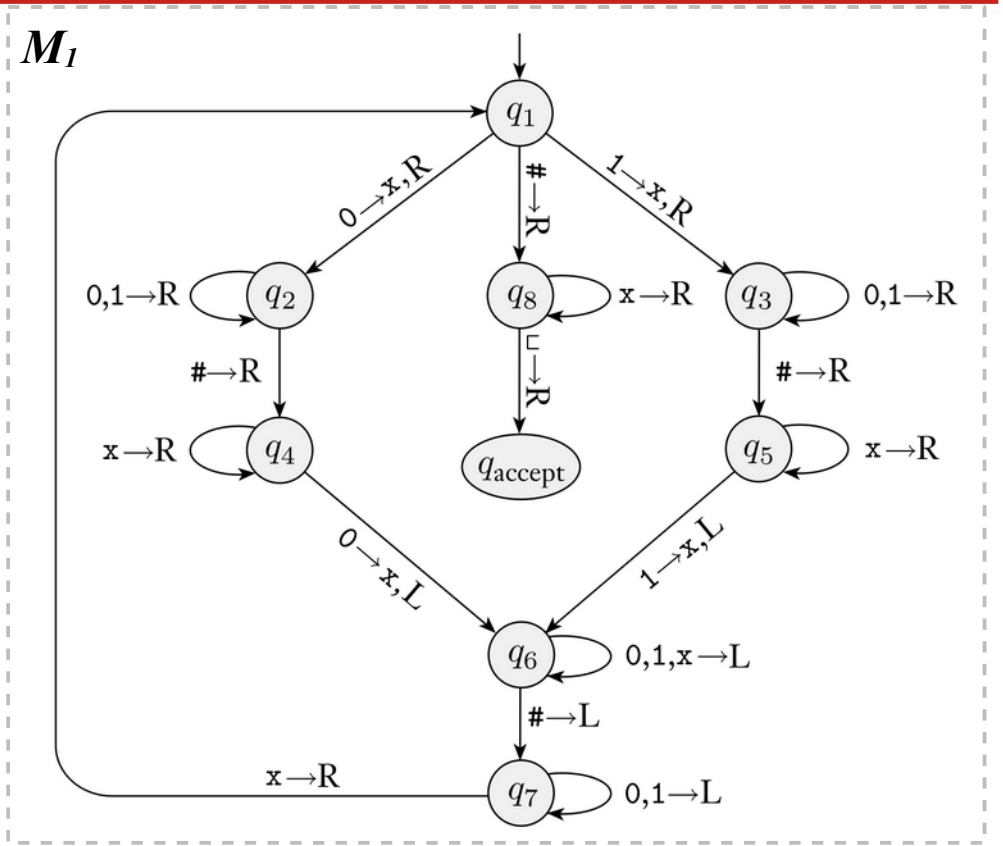
$q_1001\#001_\sqcup \Rightarrow xq_201\#001_\sqcup \Rightarrow x0q_21\#001_\sqcup \Rightarrow$
$x01\#q_4001_\sqcup \Rightarrow x01q_6\#x01_\sqcup \Rightarrow x0q_71\#x01_\sqcup \Rightarrow$
$xq_701\#x01_\sqcup \Rightarrow q_7x01\#x01_\sqcup \Rightarrow xq_101\#x01_\sqcup \Rightarrow$
$xxq_21\#x01_\sqcup \Rightarrow xx1q_2\#x01_\sqcup \Rightarrow xx1\#q_4x01_\sqcup \Rightarrow$
$xx1\#xq_401_\sqcup \Rightarrow xx1\#q_6xx1_\sqcup \Rightarrow xx1q_6\#xx1_\sqcup \Rightarrow$
$xxq_71\#xx1_\sqcup \Rightarrow xq_7x1\#xx1_\sqcup \Rightarrow xxq_11\#xx1_\sqcup \Rightarrow$
$xxxq_3\#xx1_\sqcup \Rightarrow xxx\#q_5xx1_\sqcup \Rightarrow xxx\#xq_5x1_\sqcup \Rightarrow$
$xxx\#xxq_51_\sqcup \Rightarrow xxx\#xq_6xx \Rightarrow xxx\#q_6xxx \Rightarrow$
$xxxq_6\#xxx \Rightarrow$



Turing Machines: Example

Consider $w = 001\#001$

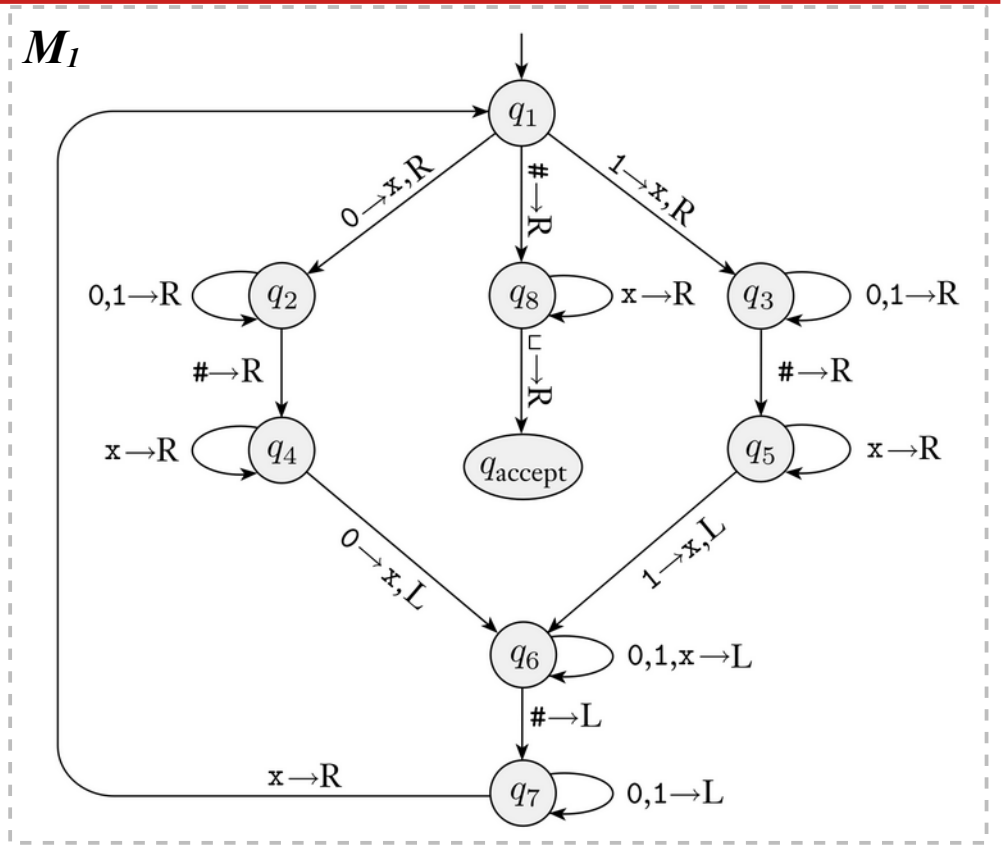
$q_1001\#001_\sqcup \Rightarrow xq_201\#001_\sqcup \Rightarrow x0q_21\#001_\sqcup \Rightarrow$
$x01\#q_4001_\sqcup \Rightarrow x01q_6\#x01_\sqcup \Rightarrow x0q_71\#x01_\sqcup \Rightarrow$
$xq_701\#x01_\sqcup \Rightarrow q_7x01\#x01_\sqcup \Rightarrow xq_101\#x01_\sqcup \Rightarrow$
$xxq_21\#x01_\sqcup \Rightarrow xx1q_2\#x01_\sqcup \Rightarrow xx1\#q_4x01_\sqcup \Rightarrow$
$xx1\#xq_401_\sqcup \Rightarrow xx1\#q_6xx1_\sqcup \Rightarrow xx1q_6\#xx1_\sqcup \Rightarrow$
$xxq_71\#xx1_\sqcup \Rightarrow xq_7x1\#xx1_\sqcup \Rightarrow xxq_11\#xx1_\sqcup \Rightarrow$
$xxxq_3\#xx1_\sqcup \Rightarrow xxx\#q_5xx1_\sqcup \Rightarrow xxx\#xq_5x1_\sqcup \Rightarrow$
$xxx\#xxq_51_\sqcup \Rightarrow xxx\#xq_6xx \Rightarrow xxx\#q_6xxx \Rightarrow$
$xxxq_6\#xxx \Rightarrow xxq_7x\#xxx \Rightarrow$



Turing Machines: Example

Consider $w = 001\#001$

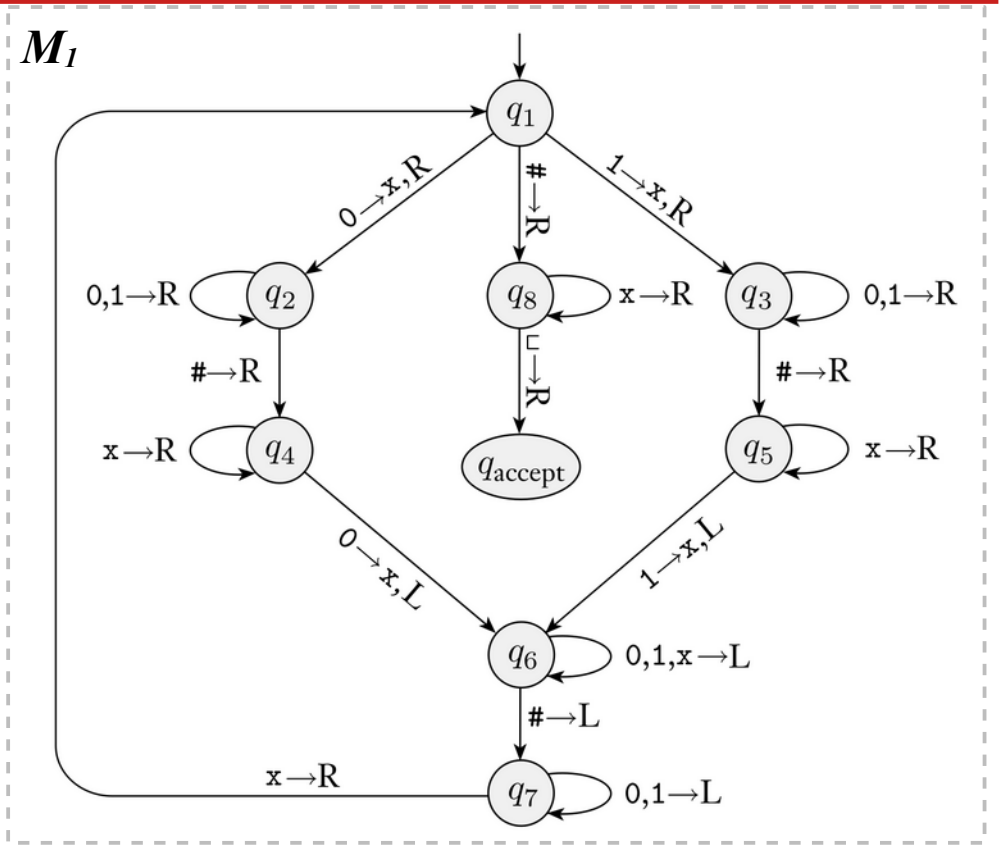
$q_1001\#001_\sqcup \Rightarrow xq_201\#001_\sqcup \Rightarrow x0q_21\#001_\sqcup \Rightarrow$
$x01\#q_4001_\sqcup \Rightarrow x01q_6\#x01_\sqcup \Rightarrow x0q_71\#x01_\sqcup \Rightarrow$
$xq_701\#x01_\sqcup \Rightarrow q_7x01\#x01_\sqcup \Rightarrow xq_101\#x01_\sqcup \Rightarrow$
$xxq_21\#x01_\sqcup \Rightarrow xx1q_2\#x01_\sqcup \Rightarrow xx1\#q_4x01_\sqcup \Rightarrow$
$xx1\#xq_401_\sqcup \Rightarrow xx1\#q_6xx1_\sqcup \Rightarrow xx1q_6\#xx1_\sqcup \Rightarrow$
$xxq_71\#xx1_\sqcup \Rightarrow xq_7x1\#xx1_\sqcup \Rightarrow xxq_11\#xx1_\sqcup \Rightarrow$
$xxxq_3\#xx1_\sqcup \Rightarrow xxx\#q_5xx1_\sqcup \Rightarrow xxx\#xq_5x1_\sqcup \Rightarrow$
$xxx\#xxq_51_\sqcup \Rightarrow xxx\#xq_6xx \Rightarrow xxx\#q_6xxx \Rightarrow$
$xxxq_6\#xxx \Rightarrow xxq_7x\#xxx \Rightarrow xxxq_1\#xxx \Rightarrow$



Turing Machines: Example

Consider $w = 001\#001$

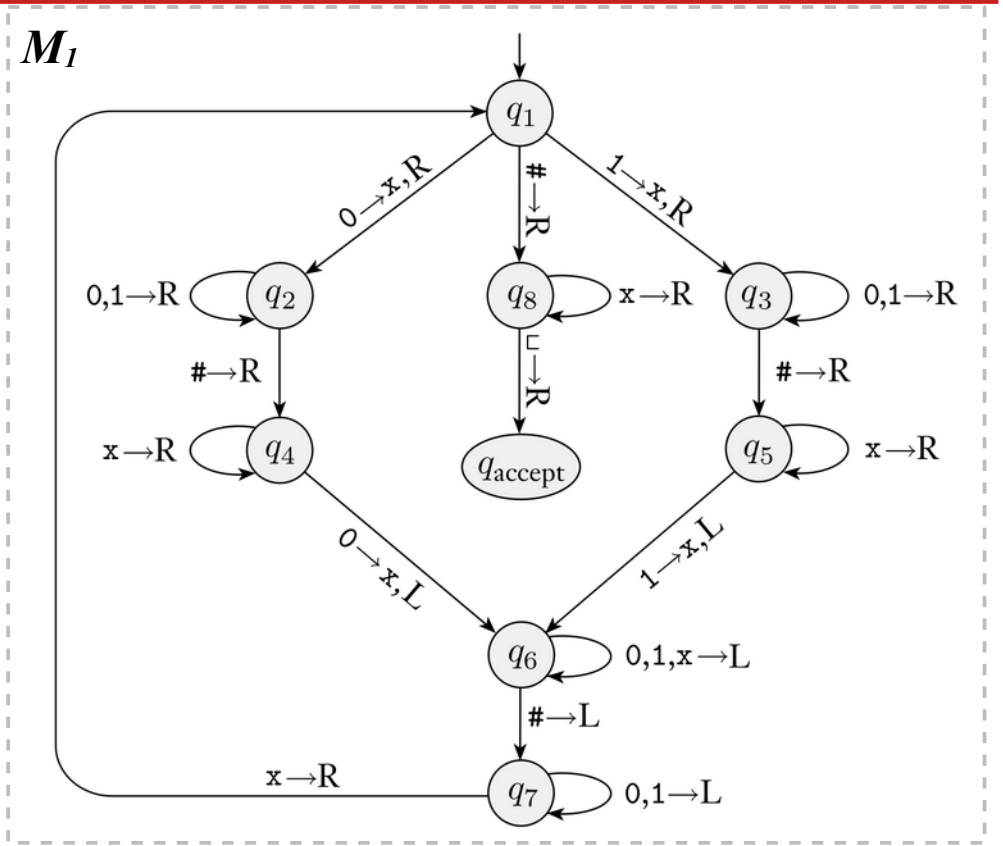
$q_1001\#001_\sqcup \Rightarrow xq_201\#001_\sqcup \Rightarrow x0q_21\#001_\sqcup \Rightarrow$
$x01\#q_4001_\sqcup \Rightarrow x01q_6\#x01_\sqcup \Rightarrow x0q_71\#x01_\sqcup \Rightarrow$
$xq_701\#x01_\sqcup \Rightarrow q_7x01\#x01_\sqcup \Rightarrow xq_101\#x01_\sqcup \Rightarrow$
$xxq_21\#x01_\sqcup \Rightarrow xx1q_2\#x01_\sqcup \Rightarrow xx1\#q_4x01_\sqcup \Rightarrow$
$xx1\#xq_401_\sqcup \Rightarrow xx1\#q_6xx1_\sqcup \Rightarrow xx1q_6\#xx1_\sqcup \Rightarrow$
$xxq_71\#xx1_\sqcup \Rightarrow xq_7x1\#xx1_\sqcup \Rightarrow xxq_11\#xx1_\sqcup \Rightarrow$
$xxxq_3\#xx1_\sqcup \Rightarrow xxx\#q_5xx1_\sqcup \Rightarrow xxx\#xq_5x1_\sqcup \Rightarrow$
$xxx\#xxq_51_\sqcup \Rightarrow xxx\#xq_6xx \Rightarrow xxx\#q_6xxx \Rightarrow$
$xxxq_6\#xxx \Rightarrow xxq_7x\#xxx \Rightarrow xxxq_1\#xxx \Rightarrow$
$xxx\#q_8xxx \Rightarrow$



Turing Machines: Example

Consider $w = 001\#001$

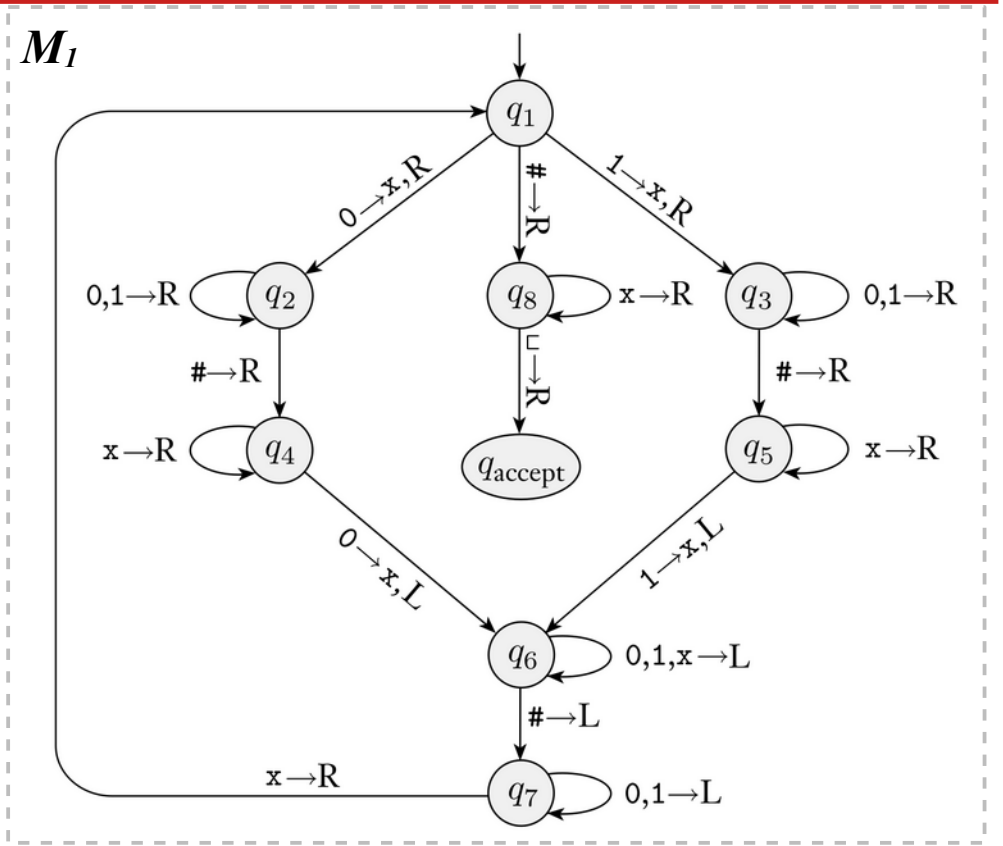
$q_1001\#001_\sqcup \Rightarrow xq_201\#001_\sqcup \Rightarrow x0q_21\#001_\sqcup \Rightarrow$
$x01\#q_4001_\sqcup \Rightarrow x01q_6\#x01_\sqcup \Rightarrow x0q_71\#x01_\sqcup \Rightarrow$
$xq_701\#x01_\sqcup \Rightarrow q_7x01\#x01_\sqcup \Rightarrow xq_101\#x01_\sqcup \Rightarrow$
$xxq_21\#x01_\sqcup \Rightarrow xx1q_2\#x01_\sqcup \Rightarrow xx1\#q_4x01_\sqcup \Rightarrow$
$xx1\#xq_401_\sqcup \Rightarrow xx1\#q_6xx1_\sqcup \Rightarrow xx1q_6\#xx1_\sqcup \Rightarrow$
$xxq_71\#xx1_\sqcup \Rightarrow xq_7x1\#xx1_\sqcup \Rightarrow xxq_11\#xx1_\sqcup \Rightarrow$
$xxxq_3\#xx1_\sqcup \Rightarrow xxx\#q_5xx1_\sqcup \Rightarrow xxx\#xq_5x1_\sqcup \Rightarrow$
$xxx\#xxq_51_\sqcup \Rightarrow xxx\#xq_6xx \Rightarrow xxx\#q_6xxx \Rightarrow$
$xxxq_6\#xxx \Rightarrow xxq_7x\#xxx \Rightarrow xxxq_1\#xxx \Rightarrow$
$xxx\#q_8xxx \Rightarrow xxx\#xq_8xx \Rightarrow$



Turing Machines: Example

Consider $w = 001\#001$

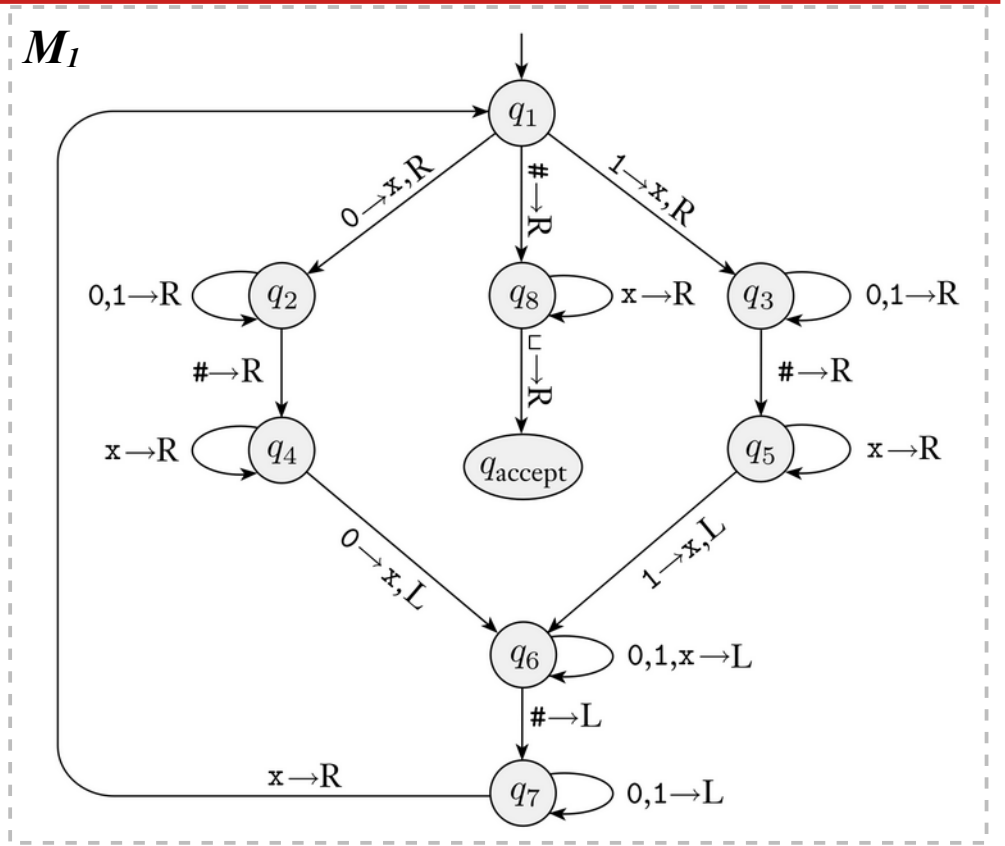
$q_1001\#001_\sqcup \Rightarrow xq_201\#001_\sqcup \Rightarrow x0q_21\#001_\sqcup \Rightarrow$
$x01\#q_4001_\sqcup \Rightarrow x01q_6\#x01_\sqcup \Rightarrow x0q_71\#x01_\sqcup \Rightarrow$
$xq_701\#x01_\sqcup \Rightarrow q_7x01\#x01_\sqcup \Rightarrow xq_101\#x01_\sqcup \Rightarrow$
$xxq_21\#x01_\sqcup \Rightarrow xx1q_2\#x01_\sqcup \Rightarrow xx1\#q_4x01_\sqcup \Rightarrow$
$xx1\#xq_401_\sqcup \Rightarrow xx1\#q_6xx1_\sqcup \Rightarrow xx1q_6\#xx1_\sqcup \Rightarrow$
$xxq_71\#xx1_\sqcup \Rightarrow xq_7x1\#xx1_\sqcup \Rightarrow xxq_11\#xx1_\sqcup \Rightarrow$
$xxxq_3\#xx1_\sqcup \Rightarrow xxx\#q_5xx1_\sqcup \Rightarrow xxx\#xq_5x1_\sqcup \Rightarrow$
$xxx\#xxq_51_\sqcup \Rightarrow xxx\#xq_6xx \Rightarrow xxx\#q_6xxx \Rightarrow$
$xxxq_6\#xxx \Rightarrow xxq_7x\#xxx \Rightarrow xxxq_1\#xxx \Rightarrow$
$xxx\#q_8xxx \Rightarrow xxx\#xq_8xx \Rightarrow xxx\#xxq_8x_\sqcup \Rightarrow$



Turing Machines: Example

Consider $w = 001\#001$

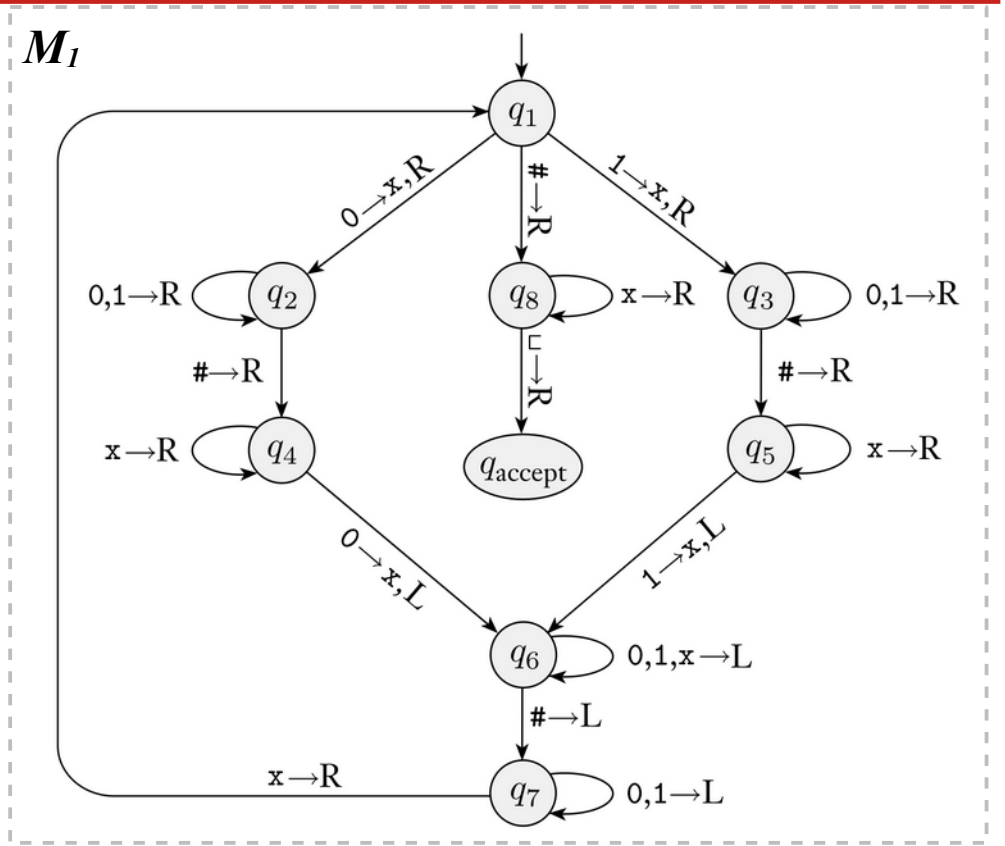
$q_1001\#001_\sqcup \Rightarrow xq_201\#001_\sqcup \Rightarrow x0q_21\#001_\sqcup \Rightarrow$
$x01\#q_4001_\sqcup \Rightarrow x01q_6\#x01_\sqcup \Rightarrow x0q_71\#x01_\sqcup \Rightarrow$
$xq_701\#x01_\sqcup \Rightarrow q_7x01\#x01_\sqcup \Rightarrow xq_101\#x01_\sqcup \Rightarrow$
$xxq_21\#x01_\sqcup \Rightarrow xx1q_2\#x01_\sqcup \Rightarrow xx1\#q_4x01_\sqcup \Rightarrow$
$xx1\#xq_401_\sqcup \Rightarrow xx1\#q_6xx1_\sqcup \Rightarrow xx1q_6\#xx1_\sqcup \Rightarrow$
$xxq_71\#xx1_\sqcup \Rightarrow xq_7x1\#xx1_\sqcup \Rightarrow xxq_11\#xx1_\sqcup \Rightarrow$
$xxxq_3\#xx1_\sqcup \Rightarrow xxx\#q_5xx1_\sqcup \Rightarrow xxx\#xq_5x1_\sqcup \Rightarrow$
$xxx\#xxq_51_\sqcup \Rightarrow xxx\#xq_6xx \Rightarrow xxx\#q_6xxx \Rightarrow$
$xxxq_6\#xxx \Rightarrow xxq_7x\#xxx \Rightarrow xxxq_1\#xxx \Rightarrow$
$xxx\#q_8xxx \Rightarrow xxx\#xq_8xx \Rightarrow xxx\#xxq_8x_\sqcup \Rightarrow$
$xxx\#xxxq_8{}_\sqcup \Rightarrow$



Turing Machines: Example

Consider $w = 001\#001$

$q_1001\#001_\sqcup \Rightarrow xq_201\#001_\sqcup \Rightarrow x0q_21\#001_\sqcup \Rightarrow$
$x01\#q_4001_\sqcup \Rightarrow x01q_6\#x01_\sqcup \Rightarrow x0q_71\#x01_\sqcup \Rightarrow$
$xq_701\#x01_\sqcup \Rightarrow q_7x01\#x01_\sqcup \Rightarrow xq_101\#x01_\sqcup \Rightarrow$
$xxq_21\#x01_\sqcup \Rightarrow xx1q_2\#x01_\sqcup \Rightarrow xx1\#q_4x01_\sqcup \Rightarrow$
$xx1\#xq_401_\sqcup \Rightarrow xx1\#q_6xx1_\sqcup \Rightarrow xx1q_6\#xx1_\sqcup \Rightarrow$
$xxq_71\#xx1_\sqcup \Rightarrow xq_7x1\#xx1_\sqcup \Rightarrow xxq_11\#xx1_\sqcup \Rightarrow$
$xxxq_3\#xx1_\sqcup \Rightarrow xxx\#q_5xx1_\sqcup \Rightarrow xxx\#xq_5x1_\sqcup \Rightarrow$
$xxx\#xxq_51_\sqcup \Rightarrow xxx\#xq_6xx \Rightarrow xxx\#q_6xxx \Rightarrow$
$xxxq_6\#xxx \Rightarrow xxq_7x\#xxx \Rightarrow xxxq_1\#xxx \Rightarrow$
$xxx\#q_8xxx \Rightarrow xxx\#xq_8xx \Rightarrow xxx\#xxq_8x_\sqcup \Rightarrow$
$xxx\#xxxq_{8\sqcup} \Rightarrow xxx\#xxxq_{accept}\sqcup$



Homework

- **Exercises:**
  - **3.1, 3.2, 3.5, and 3.8a**
- **Read 3.2 and 3.3**