CS 6041
Theory of Computation

Turing machine

Kun Suo

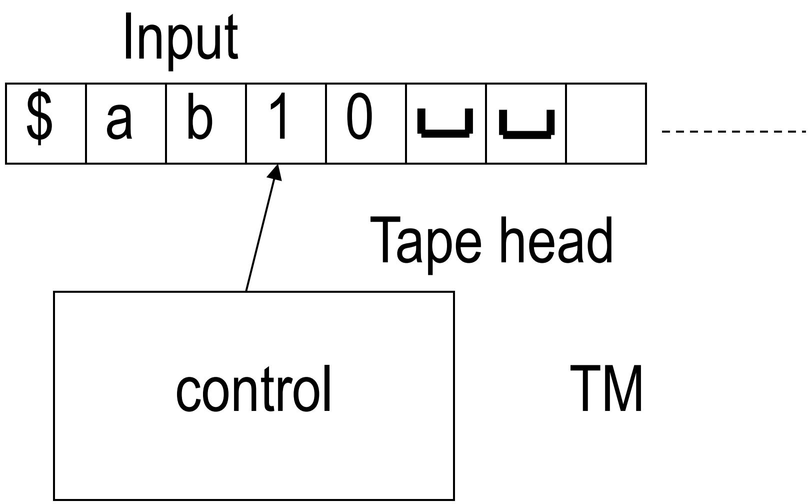
Computer Science, Kennesaw State University

https://kevinsuo.github.io/

Outline

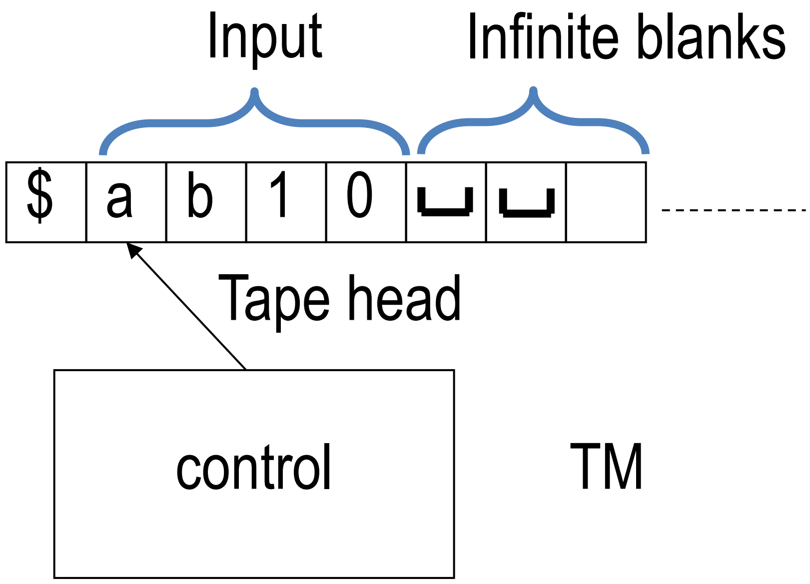
- Turing-recognizable and Turing-decidable
- Example of Turing machines
  - $\{0^{2^n}\}$
  - {w#w}
- Variants of TMs
  - Multi-tape TM
  - Nondeterministic TM

Revisit: Input on the tape of TM



- $\Sigma = \{a, b, 0,1, \ldots\}$
- $␣ \notin \Sigma$
- The blank symbol is just used to fill the infinite tape of TM

Revisit: Initial state and operations of TM



- Operations:
  - Read symbol below the head
  - Write symbol below the head
  - Move head one step left
  - Move head one step right

Revisit: The output of Turing Machine

- Accept
- Reject

$\left.\begin{array}{l}\text{Accept}\\ \text{Reject}\end{array}\right\}$ Halt

- Loop = Never Halt

For finite automata and pushdown automata, they will halt

Turing-recognizable and Turing-decidable

- Turing-recognizable: A=L(M)
  - $x \in A$, M accept x
  - $x \notin A$, M reject x or loop
- Turing-decidable: A=L(M)
  - $x \in A$, M accept x
  - $x \notin A$, M reject x

Halt no matter what is the input

- Turing-recognizable $\neq$ Turing-decidable

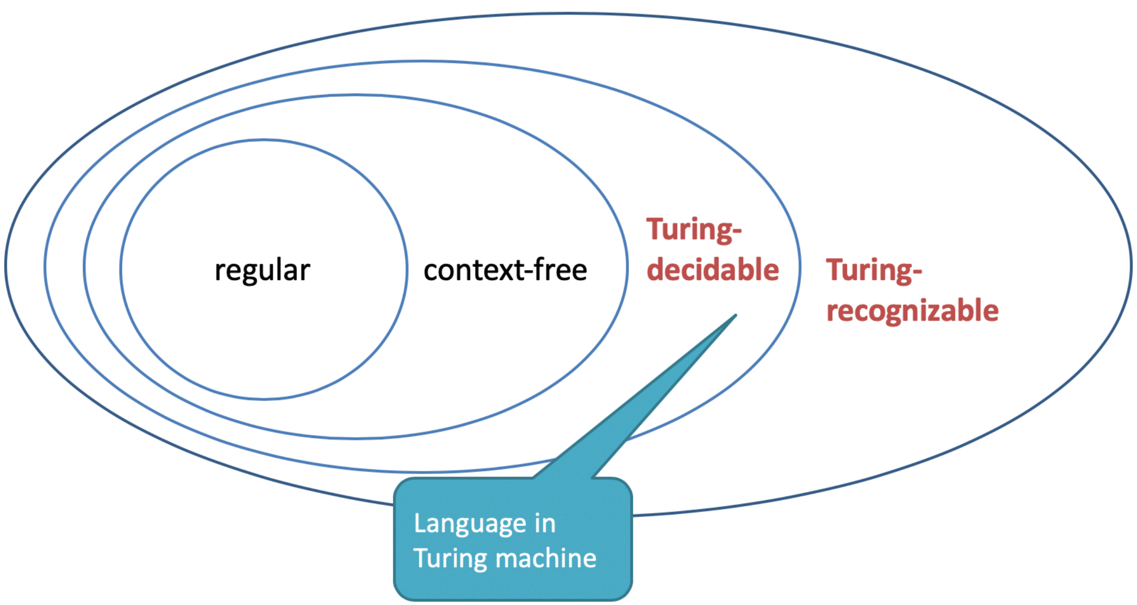
Revisit: The output of Turing Machine

- Accept
- Reject
- Loop

Accept, Reject } Halt -> Decidable

Loop = Never Halt

Halt -> Decidable, Never Halt } Recognizable



Example of TM: A = $\{0^{2^n} \mid n \geq 0\}$

- Prove: A is not regular.

Suppose A is regular language and p is its pumping length

S = $0^{2^p} \in A$, |S|=2^p > p, so S = xyz

$|y|>0$

$|xy| \leq p$

$\forall i \geq 0, xy^i z \in A$

Suppose |xy| = k $\leq p < 2^p$, then |z| = 2^p – k > 0

For string xyyz, the length $2^p < |xyyz| \leq 2^p + k < 2^p + 2^p = 2^{p+1}$

So |xyyz| is not in A, contradiction!

Example of TM: A = $\{0^{2^n} \mid n \geq 0\}$

- Prove A is not context-free.

Suppose A is CFL and p is its pumping length

S = $0^{2^p} \in A$, |S|=2^p > p, so S = uvxyz

$\forall i \geq 0$, $uv^ixy^iz \in A$;

|vy|>0;

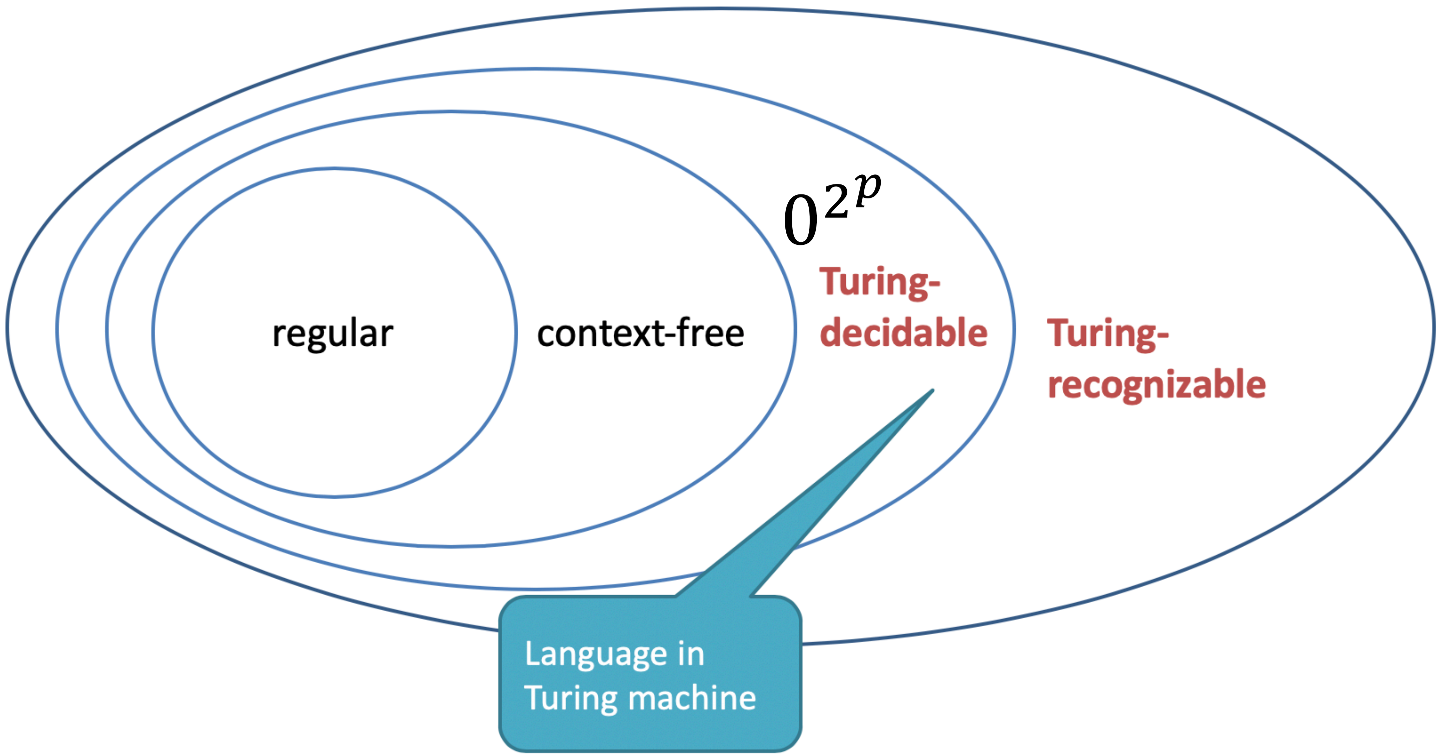
|vxy|$\leq$p.

Suppose |vxy| = k $\leq p < 2^p$, then |uz| = 2^p – k > 0

For string uv^2xy^2z, the length $2^p < |uv^2xy^2z| \leq 2^p + k < 2^p + 2^p = 2^{p+1}$

So $|uv^2xy^2z|$ is not in A, contradiction!

Example of TM: A = $\{0^{2^n} \mid n \geq 0\}$



Example of TM: A = $\{0^{2^n} \mid n \geq 0\}$

Not regular language or context free language

- M2 = "On input string w:
  1) Sweep left to right across the tape, crossing off every other 0. e.g., $\underline{0}0\underline{0}0\underline{0}0\underline{0}0$
  2) If in stage 1 the tape contained a single 0, accept . N=0
  3) If in stage 1 the tape contained more than a single 0 and the number of 0s was odd, reject .
  4) Return the head to the left-hand end of the tape.
  5) Go to stage 1."

Example of TM: A = $\{0^{2^n} \mid n \geq 0\}$

- $\underline{0}0\underline{0}0\underline{0}0\underline{0}0$

$\Rightarrow$ $\underline{0}0\underline{0}0$

$\Rightarrow$ $\underline{0}0$

$\Rightarrow$ 0, accept

- $\underline{0}0\underline{0}0\underline{0}0\underline{0}0\underline{0}0$

$\Rightarrow$ $\underline{0}0\underline{0}0\underline{0}$,

$\Rightarrow$ odd length, reject

- M2 = "On input string w:

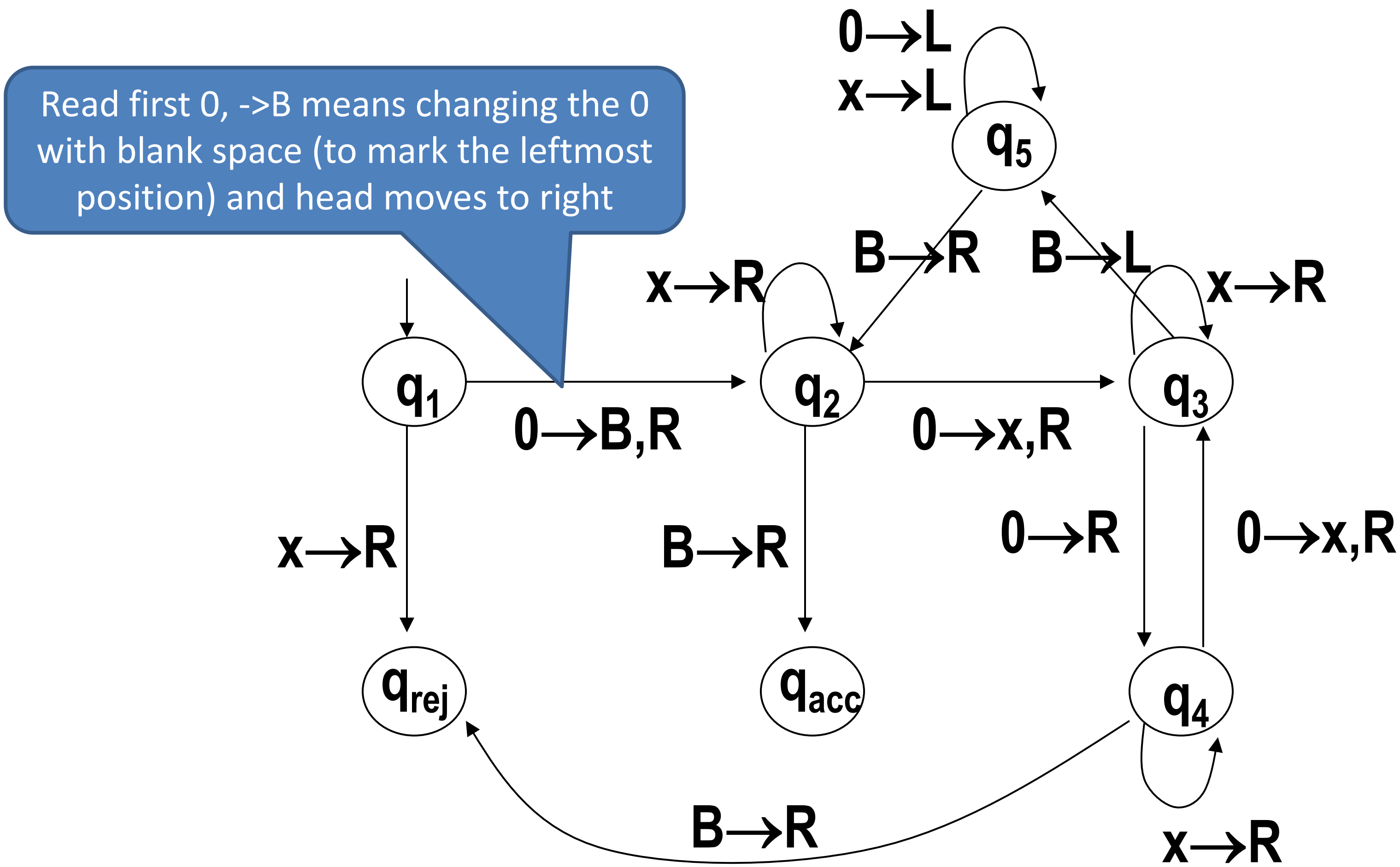
1) Sweep left to right across the tape, crossing off every other 0.
2) If in stage 1 the tape contained a single 0, accept .
3) If in stage 1 the tape contained more than a single 0 and the number of 0s was odd, reject .
4) Return the head to the left-hand end of the tape.
5) Go to stage 1."

Example of TM: A = $\{0^{2^n} \mid n \geq 0\}$

- Create a TM for A

- $M_2=(Q,\Sigma,\Gamma,\delta,q_1,q_{acc},q_{rej})$
  - $Q=\{q_1, q_2, q_3, q_4, q_5, q_{acc}, q_{rej},\}$
  - $\Sigma=\{0\}$
  - $\Gamma=\{0,X,B\}$, B denotes space
  - δ is shown in next page

Example of TM: A = $\{0^{2^n} \mid n \geq 0\}$



Example of TM: A = $\{0^{2^n} \mid n \geq 0\}$

- M_2 :computation on 0000

  $q_1 0000$, $Bq_2 000$, $BXq_3 00$, $BX0q_4 0$, $BX0Xq_3 B$,

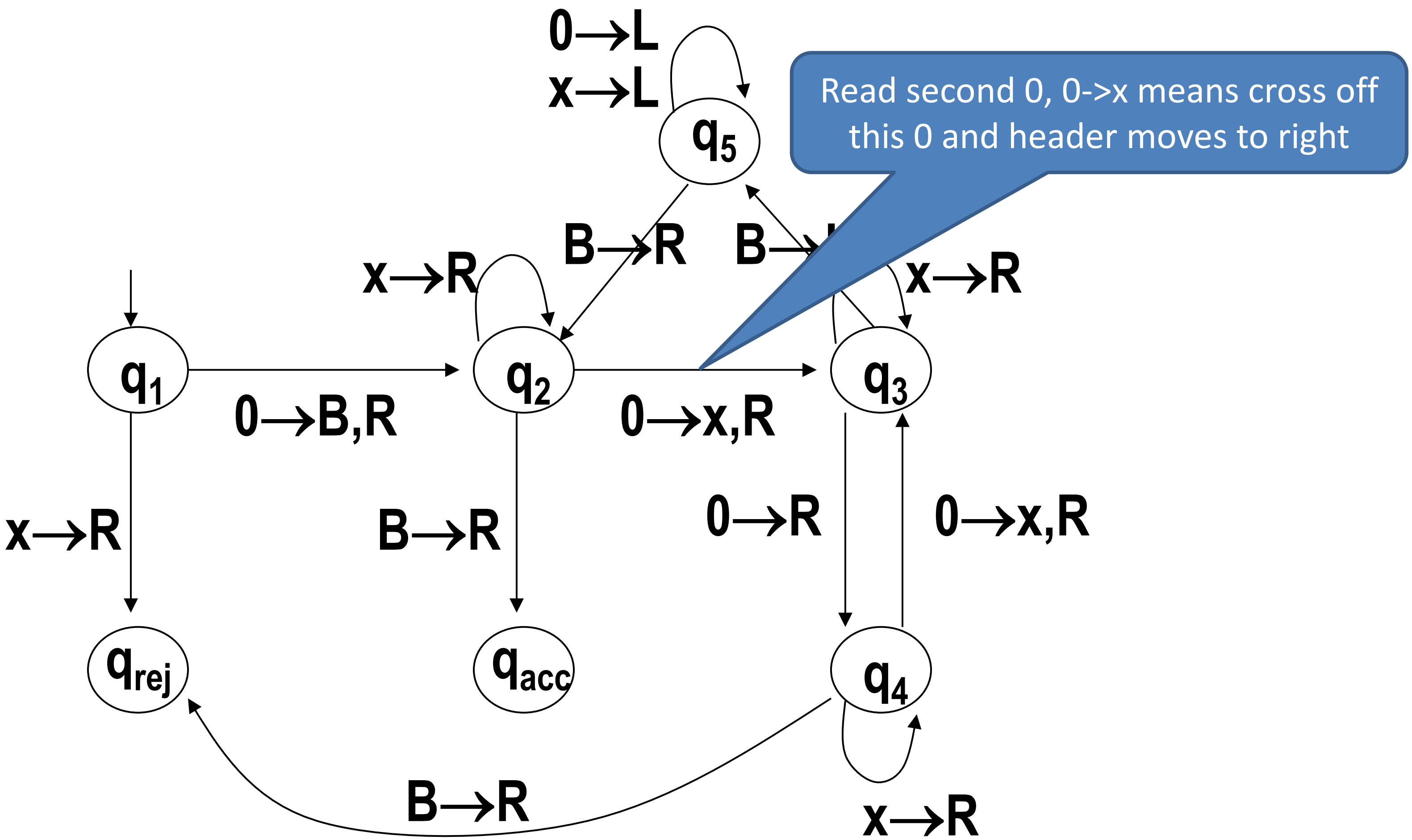
  $BX0q_5 XB$, $BXq_5 0XB$, $Bq_5 X0XB$, $q_5 BX0XB$,

  $Bq_2 X0XB$, $BXq_2 0XB$, $BXXq_3 XB$, $BXXXq_3 B$,

  $BXXq_5 XB$, $BXq_5 XXB$, $Bq_5 XXXB$, $q_5 BXXXB$,

  $Bq_2 XXXB$, $BXq_2 XXB$, $BXXq_2 XB$, $BXXXq_2 B$, $BXXXBq_{acc}$

Example of TM: A = $\{0^{2^n} \mid n \geq 0\}$



Example of TM: $A = \{0^{2^n} \mid n \geq 0\}$

- M_2 :computation on 0000

  q_10000, Bq_2000, BXq_300, $BX0q_40$, $BX0Xq_3B$,

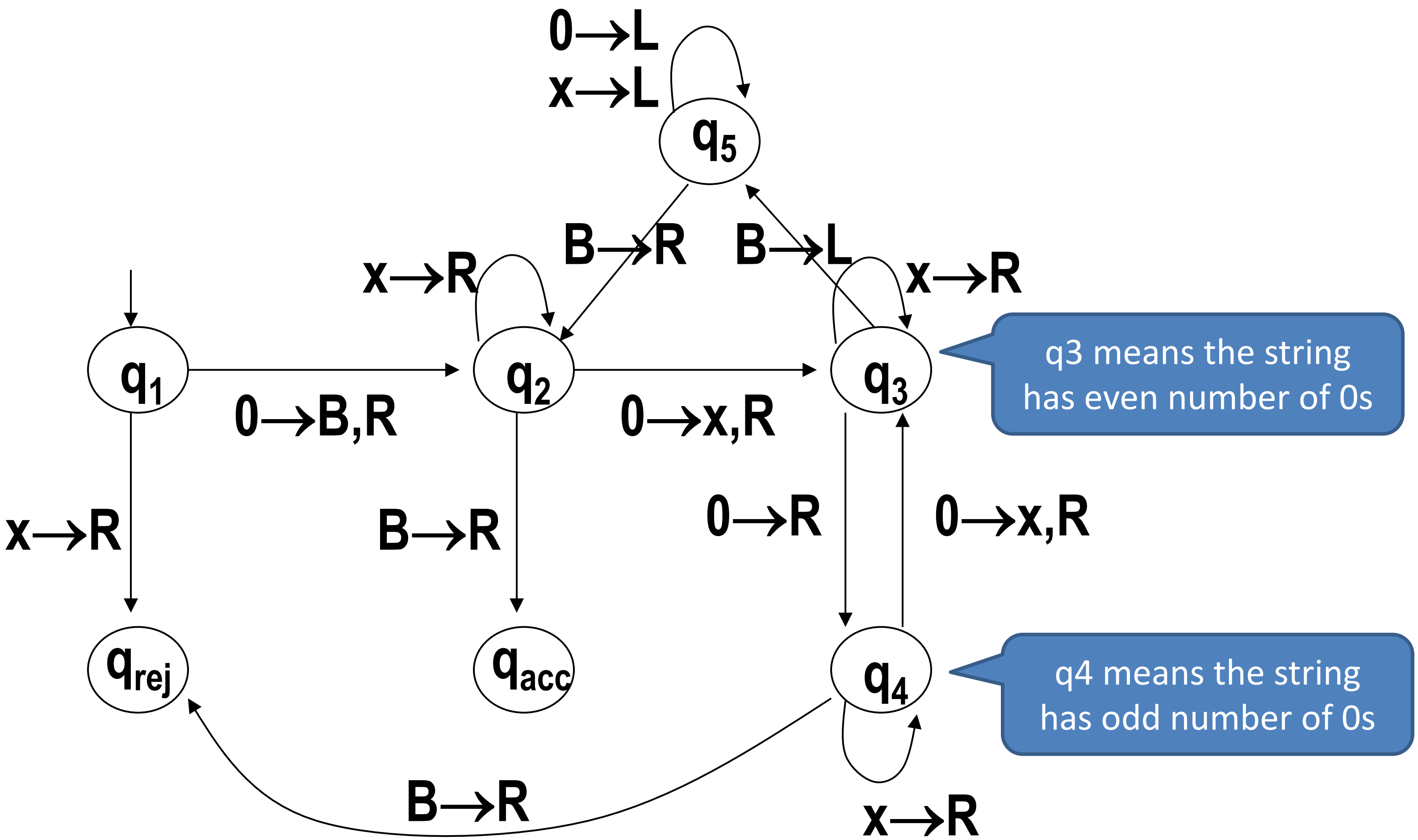
  $BX0q_5XB$, BXq_50XB, Bq_5X0XB, q_5BX0XB,

  Bq_2X0XB, BXq_20XB, $BXXq_3XB$, $BXXXq_3B$,

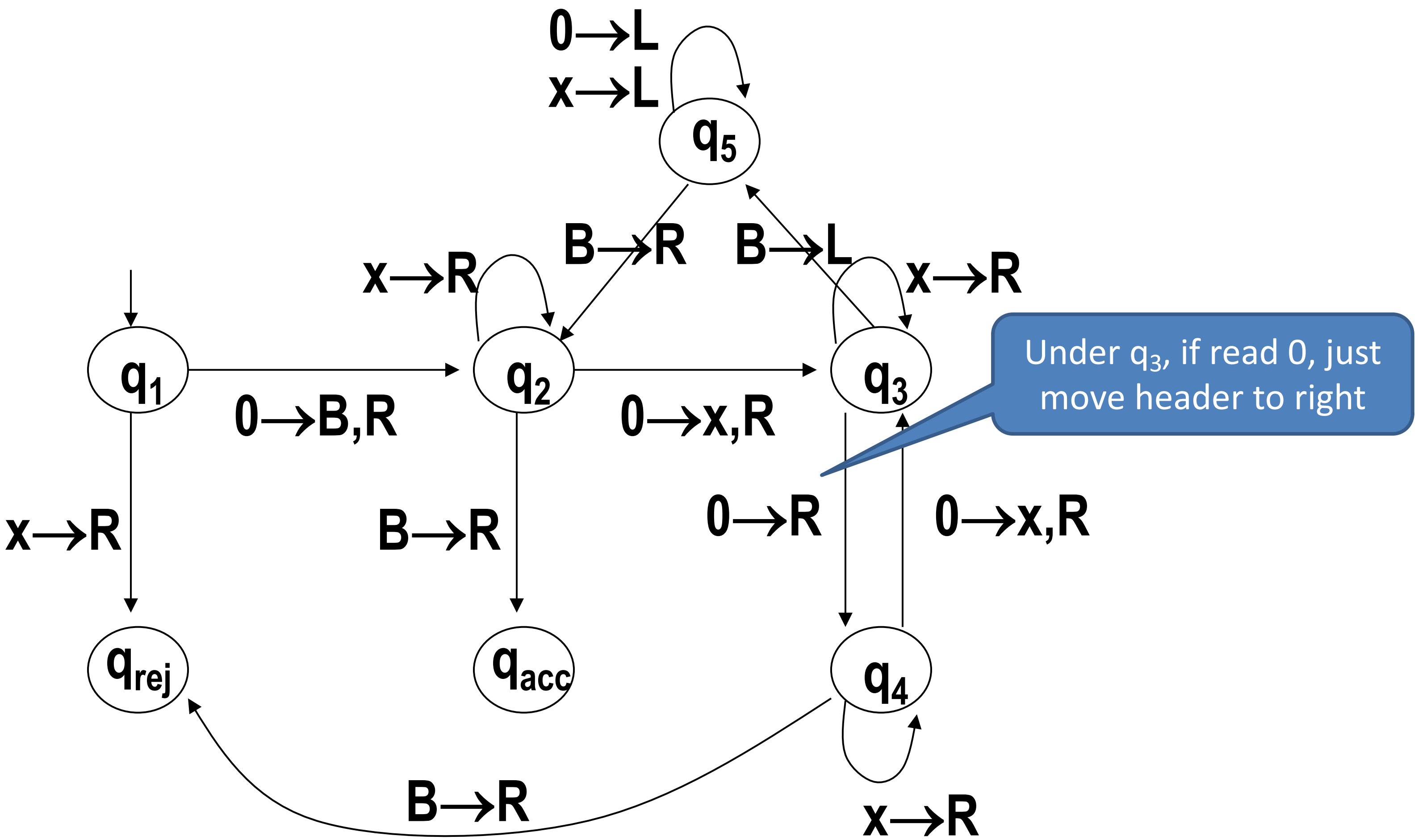
  $BXXq_5XB$, BXq_5XXB, Bq_5XXXB, q_5BXXXB,

  Bq_2XXXB, BXq_2XXB, $BXXq_2XB$, $BXXXq_2B$, $BXXXBq_{acc}$

Example of TM: $A = \{0^{2^n} \mid n \geq 0\}$



Example of TM: A = $\{0^{2^n} \mid n \geq 0\}$



Example of TM: A = $\{0^{2^n} \mid n \geq 0\}$

- M_2 :computation on 0000

  q_10000, Bq_2000, BXq_300, $BX0q_40$, $BX0Xq_3B$,

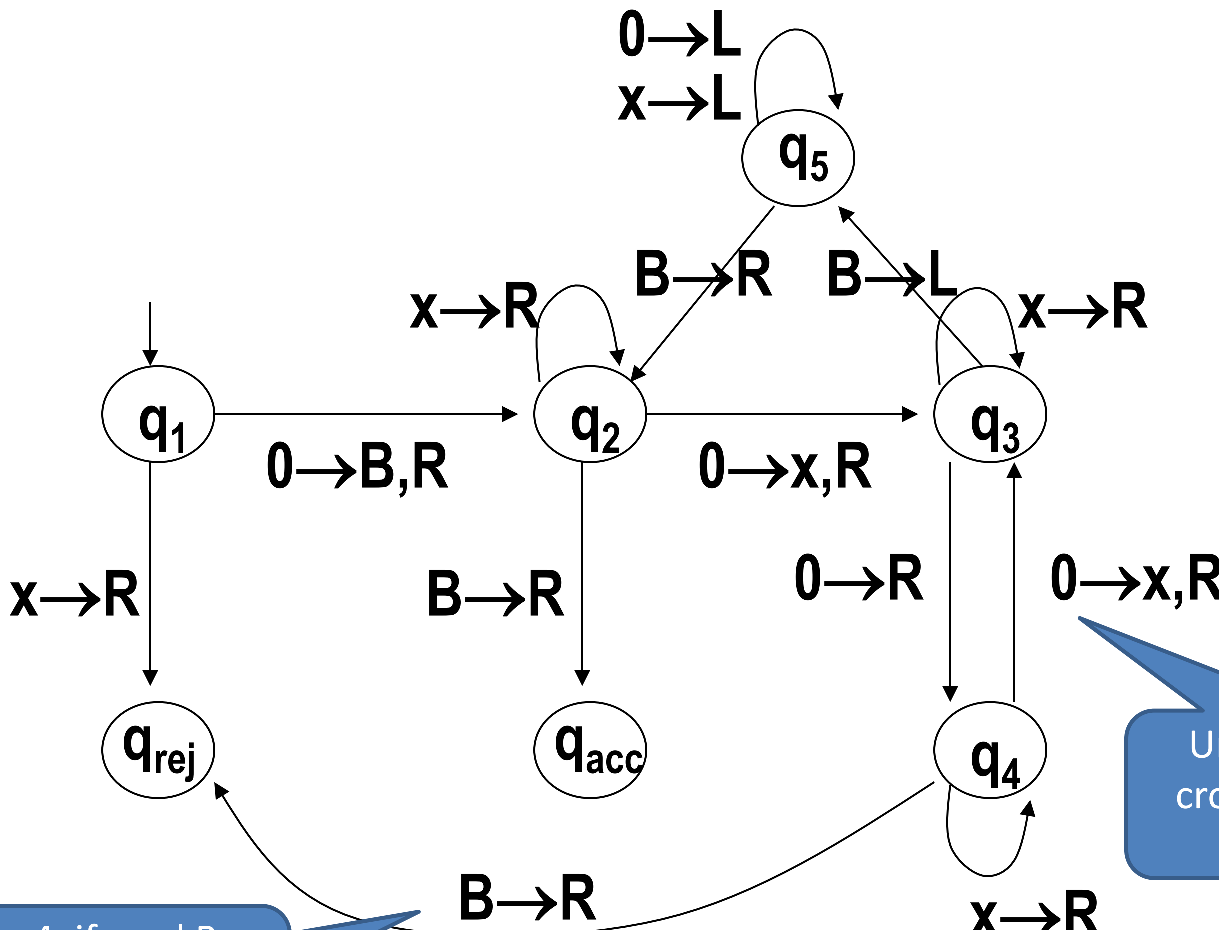
  $BX0q_5XB$, BXq_50XB, Bq_5X0XB, q_5BX0XB,

  Bq_2X0XB, BXq_20XB, $BXXq_3XB$, $BXXXq_3B$,

  $BXXq_5XB$, BXq_5XXB, Bq_5XXXB, q_5BXXXB,

  Bq_2XXXB, BXq_2XXB, $BXXq_2XB$, $BXXXq_2B$, $BXXXBq_{acc}$

Example of TM: A = $\{0^{2^n} \mid n \geq 0\}$



Under q4, if read 0, cross off it and move header to right

Under q4, if read B, means we have odd number of 0s, reject

Example of TM: A = $\{0^{2^n} \mid n \geq 0\}$

- M_2 :computation on 0000

  $q_1 0000$, $Bq_2 000$, $BXq_3 00$, $BX0q_4 0$, $BX0Xq_3 B$,

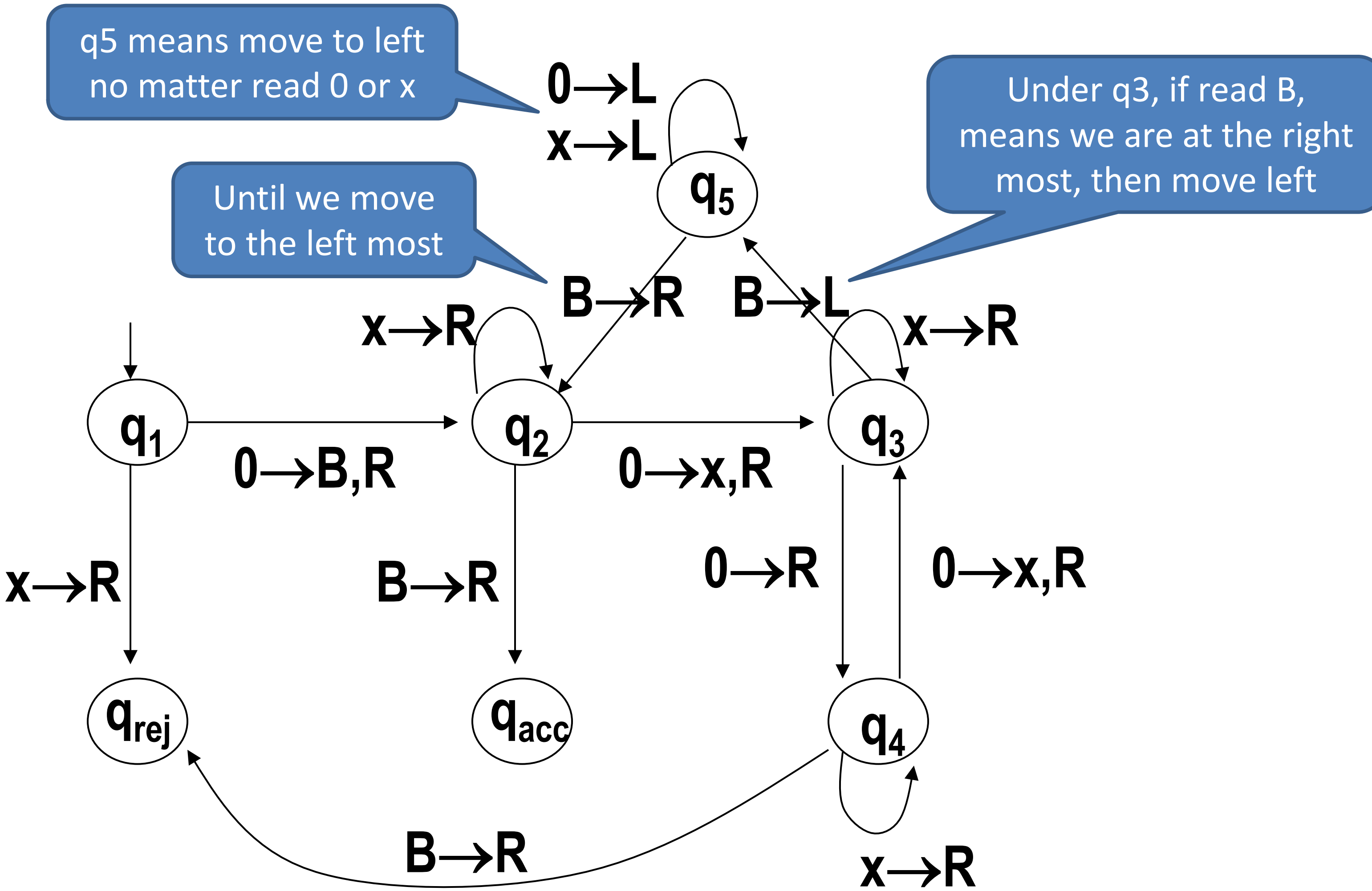
  $BX0q_5 XB$, $BXq_5 0XB$, $Bq_5 X0XB$, $q_5 BX0XB$,

  $Bq_2 X0XB$, $BXq_2 0XB$, $BXXq_3 XB$, $BXXXq_3 B$,

  $BXXq_5 XB$, $BXq_5 XXB$, $Bq_5 XXXB$, $q_5 BXXXB$,

  $Bq_2 XXXB$, $BXq_2 XXB$, $BXXq_2 XB$, $BXXXq_2 B$, $BXXXBq_{acc}$

Example of TM: A = $\{0^{2^n} \mid n \geq 0\}$



Example of TM: $A = \{0^{2^n} \mid n \geq 0\}$

- M_2 :computation on 0000

  $q_1 0000$, $Bq_2 000$, $BXq_3 00$, $BX0q_4 0$, $BX0Xq_3 B$,

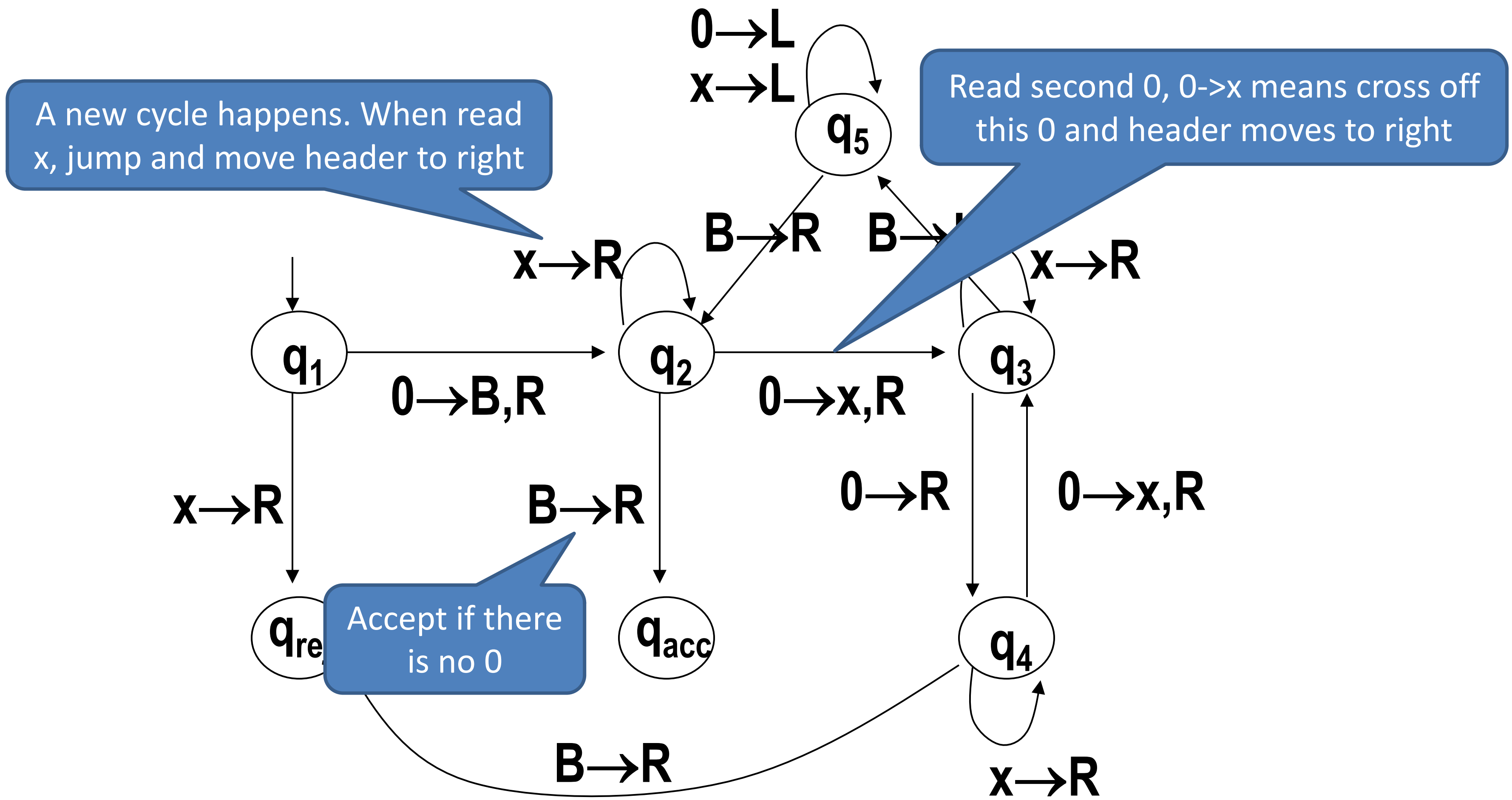
  $BX0q_5 XB$, $BXq_5 0XB$, $Bq_5 X0XB$, $q_5 BX0XB$,

  $Bq_2 X0XB$, $BXq_2 0XB$, $BXXq_3 XB$, $BXXXq_3 B$,

  $BXXq_5 XB$, $BXq_5 XXB$, $Bq_5 XXXB$, $q_5 BXXXB$,

  $Bq_2 XXXB$, $BXq_2 XXB$, $BXXq_2 XB$, $BXXXq_2 B$, $BXXXBq_{acc}$

Example of TM: A = $\{0^{2^n} \mid n \geq 0\}$



Example of TM: A = $\{0^{2^n} \mid n \geq 0\}$

- M_2 :computation on 0000

  $q_1 0000$, $Bq_2 000$, $BXq_3 00$, $BX0q_4 0$, $BX0Xq_3 B$,

  $BX0q_5 XB$, $BXq_5 0XB$, $Bq_5 X0XB$, $q_5 BX0XB$,

  $Bq_2 X0XB$, $BXq_2 0XB$, $BXXq_3 XB$, $BXXXq_3 B$,

  $BXXq_5 XB$, $BXq_5 XXB$, $Bq_5 XXXB$, $q_5 BXXXB$,

  $Bq_2 XXXB$, $BXq_2 XXB$, $BXXq_2 XB$, $BXXXq_2 B$, $BXXXBq_{acc}$

Example of TM: A = $\{0^{2^n} \mid n \geq 0\}$

- M_2 :computation on 0000

0000

$q_1 0000$, $Bq_2 000$, $BXq_3 00$, $BX0q_4 0$, $BX0Xq_3 B$, $\Rightarrow$ BX0X

$BX0q_5 XB$, $BXq_5 0XB$, $Bq_5 X0XB$, $q_5 BX0XB$, $\Leftarrow$

$Bq_2 X0XB$, $BXq_2 0XB$, $BXXq_3 XB$, $BXXXq_3 B$, $\Rightarrow$ BXXX

$BXXq_5 XB$, $BXq_5 XXB$, $Bq_5 XXXB$, $q_5 BXXXB$, $\Leftarrow$

$Bq_2 XXXB$, $BXq_2 XXB$, $BXXq_2 XB$, $BXXXq_2 B$, $BXXXBq_{acc}$ $\Rightarrow$ BXXX

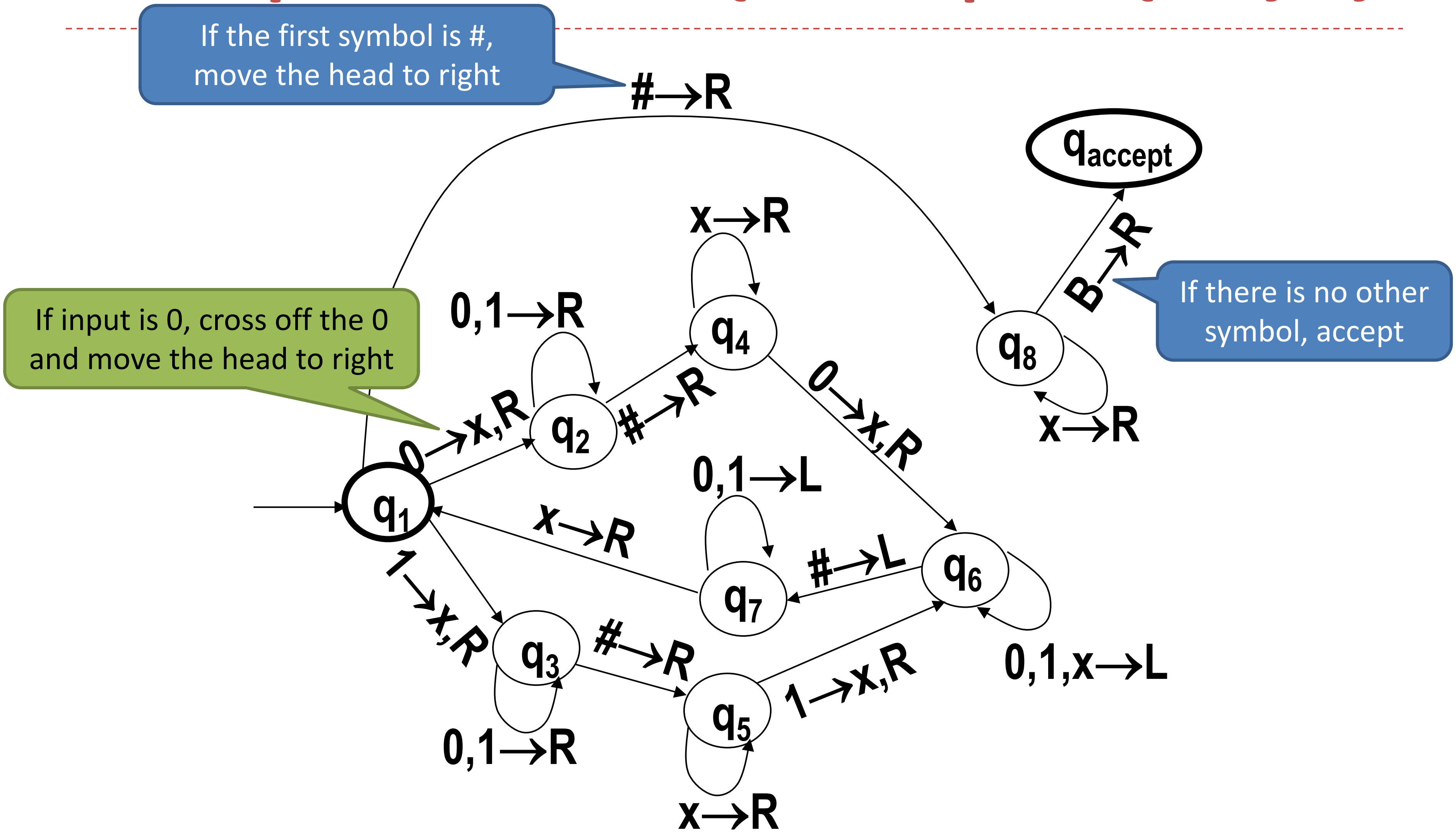
Example of TM: $B=\{ w\#w \mid w\in\{0,1\}^* \}$

- M_1= "for input string x":
  1. Scan the input to make sure there exists only one "#", otherwise reject;
  2. Move to the same positions on both sides between "#", check whether there exist same symbols. If not, reject; otherwise, cross off the checked symbols;
  3. If all symbols on the left of "#" are crossed off, check whether there exists other remaining symbols on the right. If yes, reject; otherwise, accept.

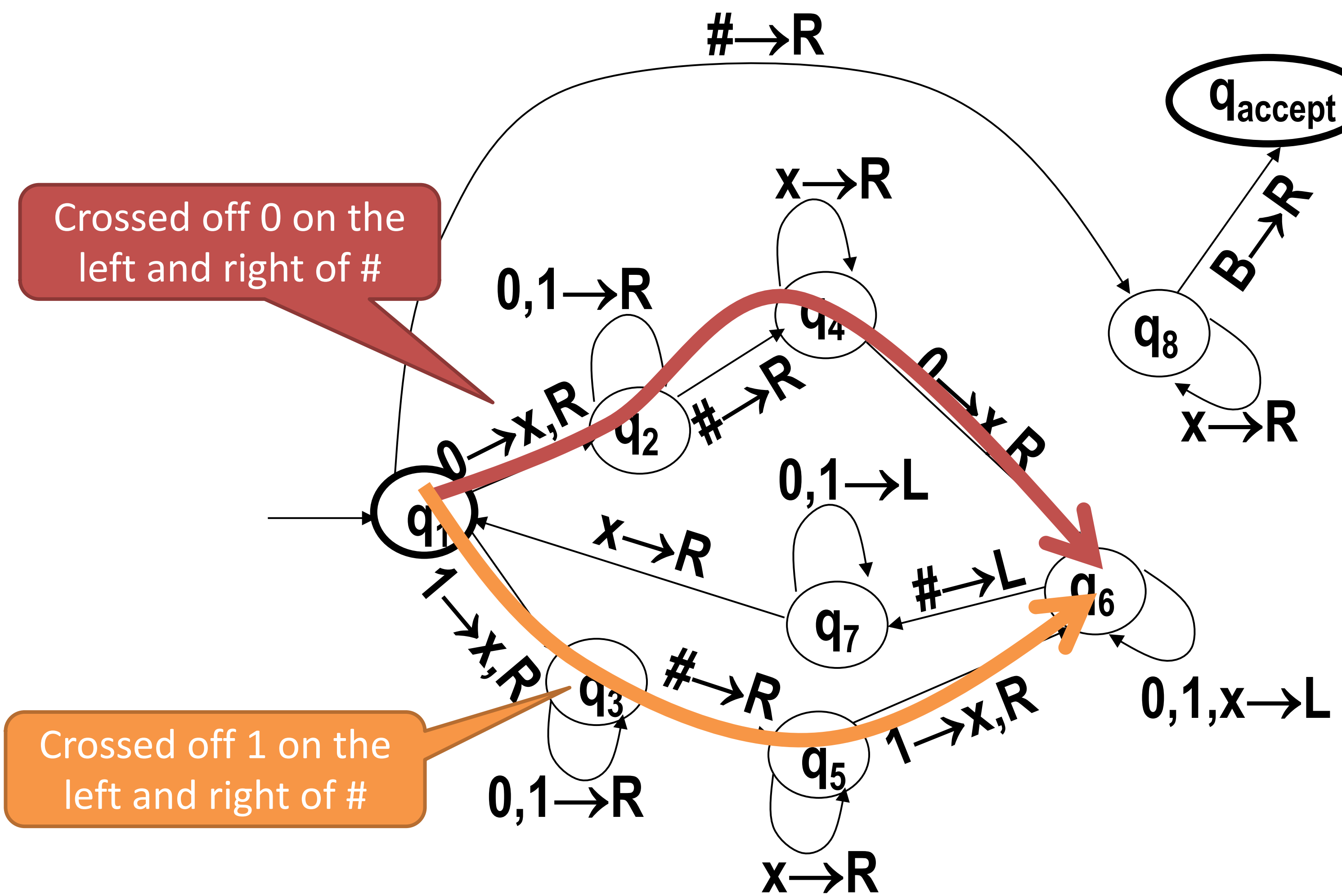
Example of TM: $B=\{ w\#w \mid w \in \{0,1\}^* \}$

- $M_1=(Q,\Sigma,\Gamma,\delta,q_1,q_{acc},q_{rej})$
  - $Q=\{q_1, q_2, \ldots\ldots, q_8, q_{acc}, q_{rej},\}$
  - $\Sigma=\{0,1,\#\}$
  - $\Gamma=\{0,1,\#,X,B\}$, B denotes space, X denotes crossed off
  - δ is shown as below: (ignore reject state)

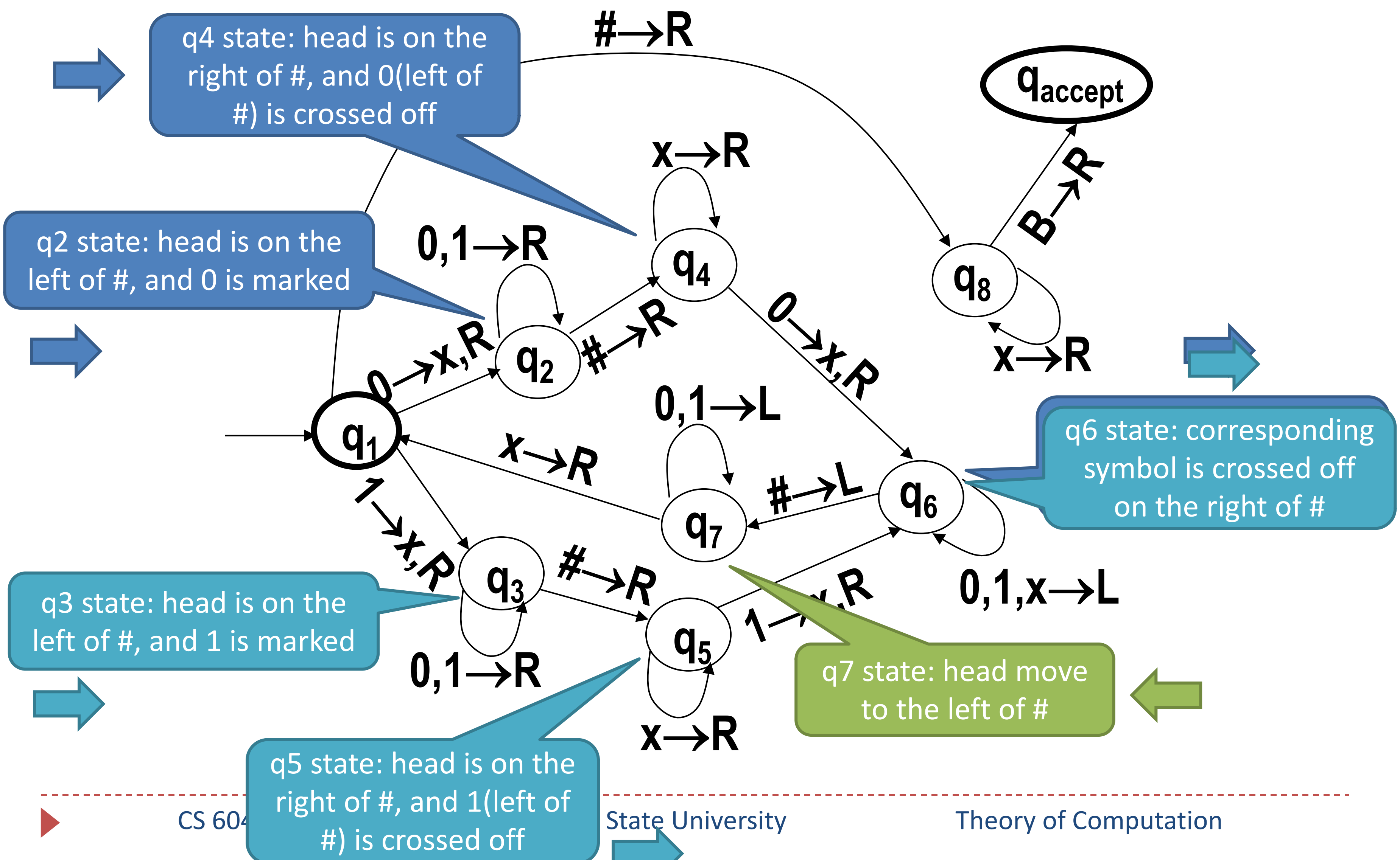
Example of TM: $B=\{ w\#w \mid w \in \{0,1\}^* \}$



Example of TM: B={ w#w | w∈{0,1}* }



Example of TM: B={ w#w | w∈{0,1}* }

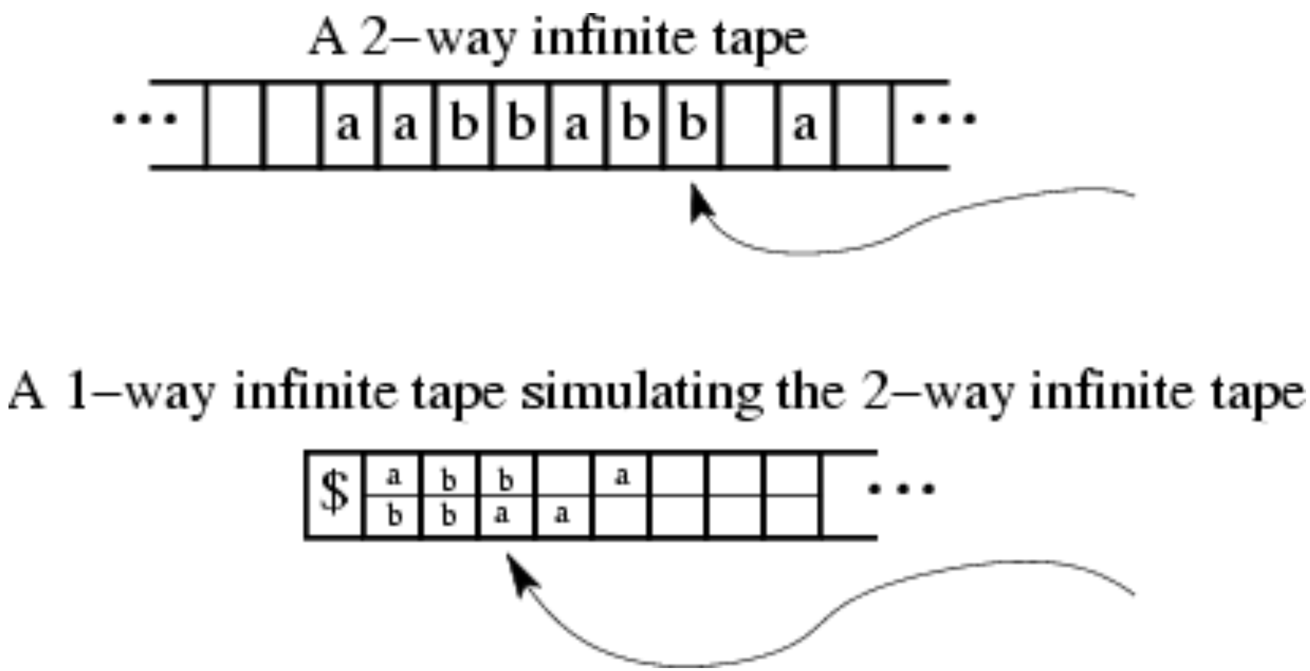


Variants of TM

- Bidirectional infinite tape Turing machine
- Multidimensional tape Turing machine
- ***Multitape Turing machine***
- Multihead Turing machine
- ***Nondeterministic Turing machine***
- ……
- TM has ***robust*** definition

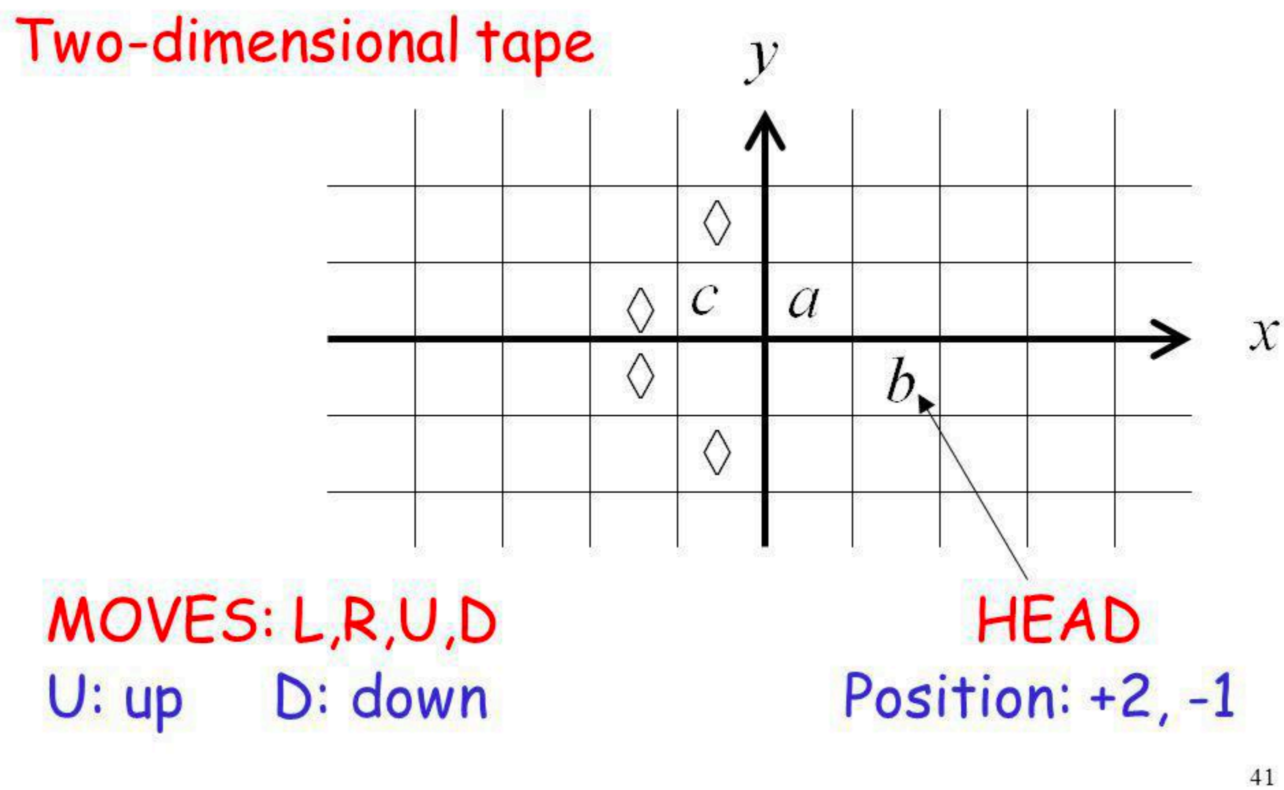
We can change the forms of TM but its functionality does not change

Variants of TM



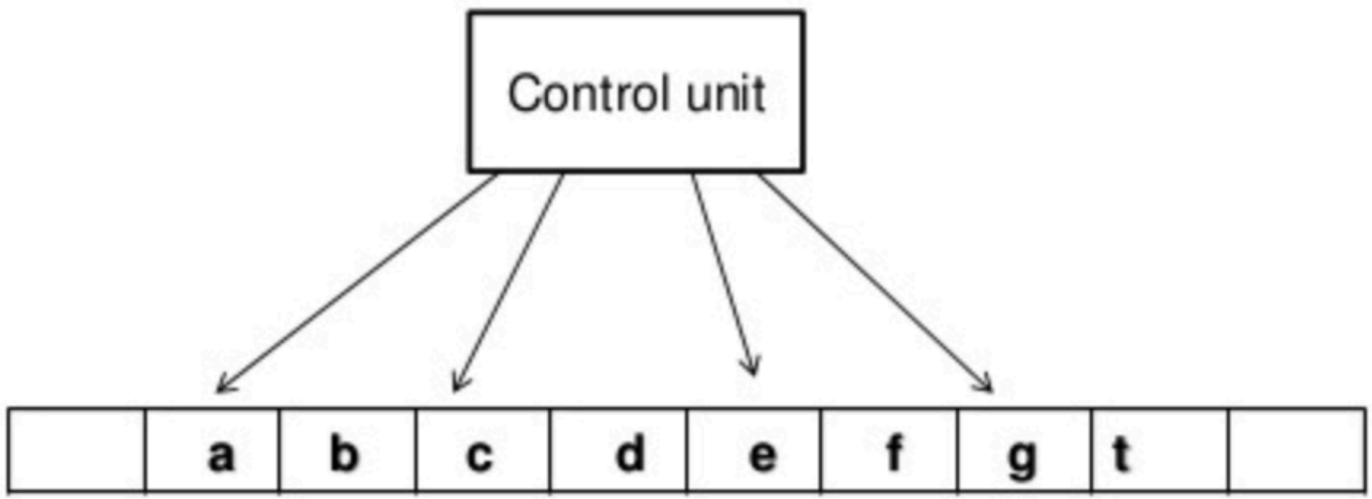
Bidirectional infinite tape Turing machine

Variants of TM



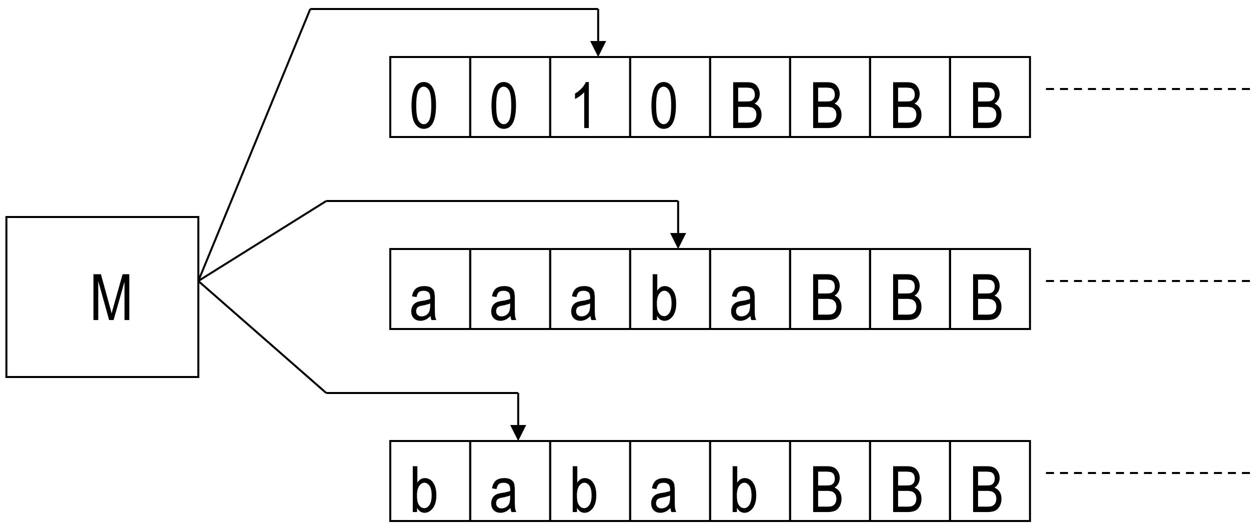
Multidimensional tape Turing machine

Variants of TM



Multi-head Turing machine

Multitape Turing machine



- **Transition function of multitape TM:**

$$\delta: Q\times\Gamma^{k}\rightarrow Q\times\Gamma^{k}\times\{L,R\}^{k}$$

k tapes

$$\delta(q_i,a_1,\ldots,a_k) = (q_j,b_1,\ldots,b_k,L,R,\ldots,L)$$

Change from a_k to b_k and move L/R

Multitape TM = single-tape TM

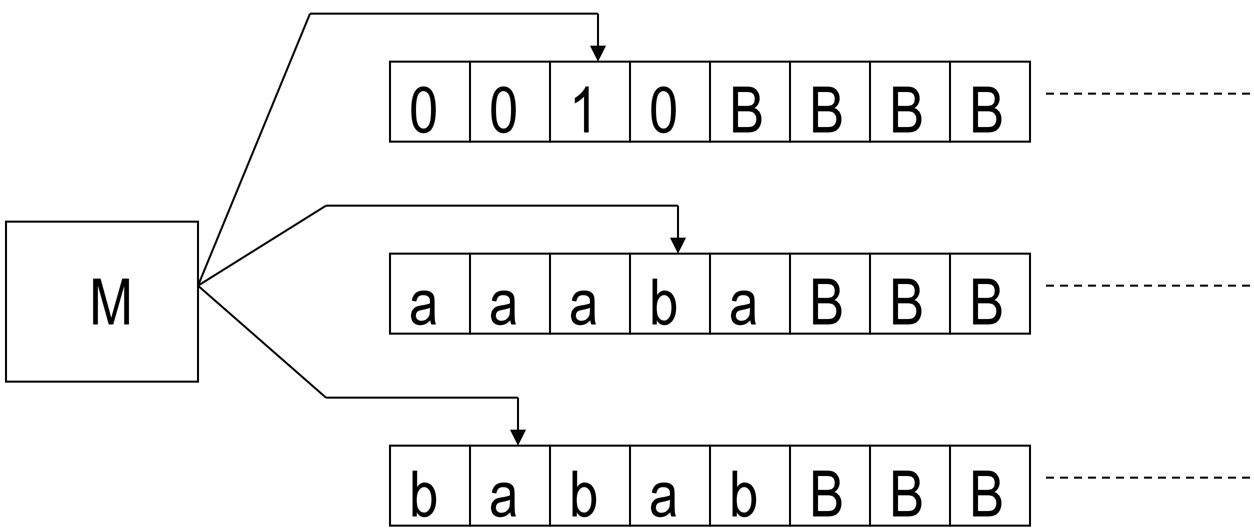
- Theorem: every multitape TM has an equivalent single-tape TM

- How to use single-tape TM to simulate multitape TM?

  (Prove by construction)

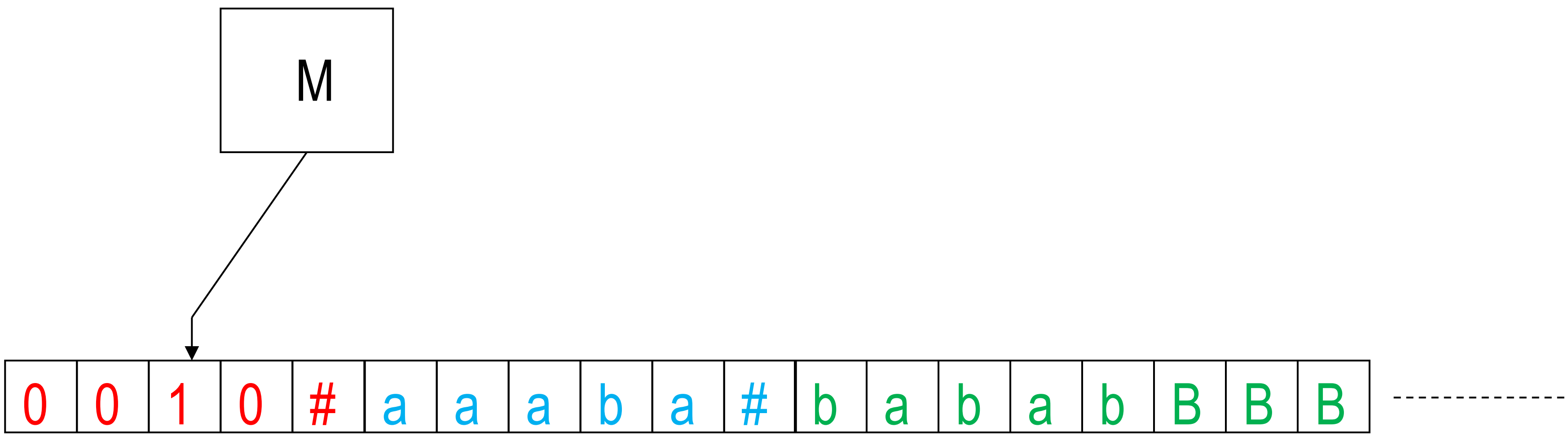
Multitape TM = single-tape TM

- How to store multi-tape on a single tape
- How to simulate multi-head on a single tape
- How to simulate one move on multi-tape on a single tape



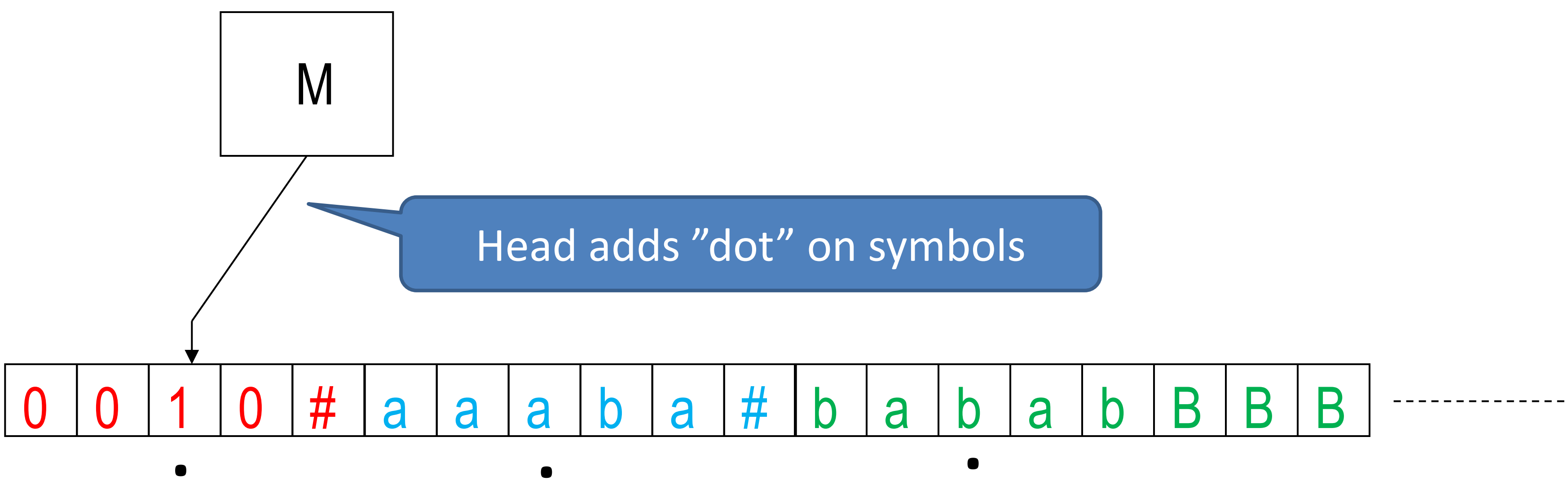
Multitape TM = single-tape TM

- **How to store multi-tape on a single tape**
- How to simulate multi-head on a single tape
- How to simulate one move on multi-tape on a single tape

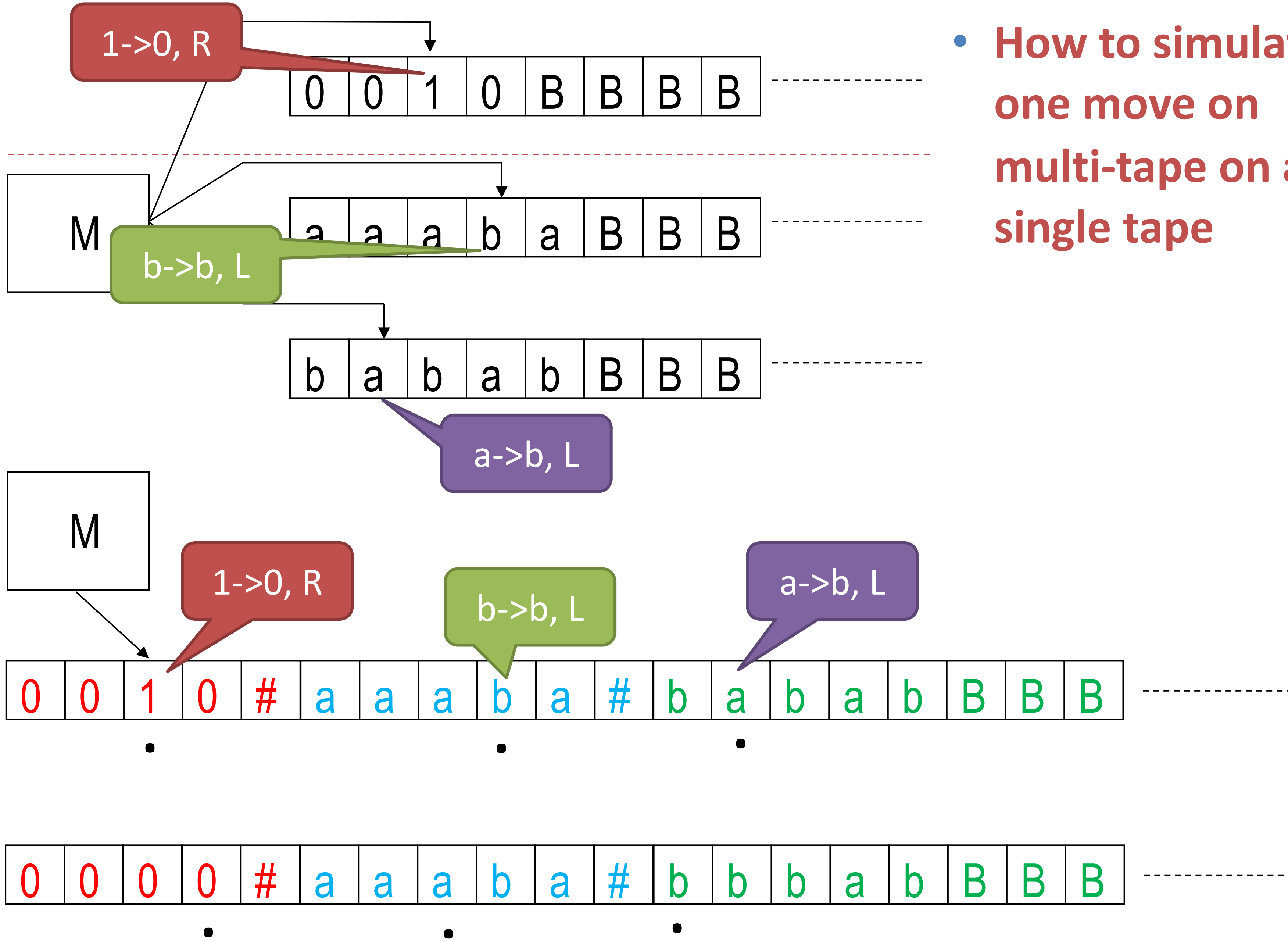


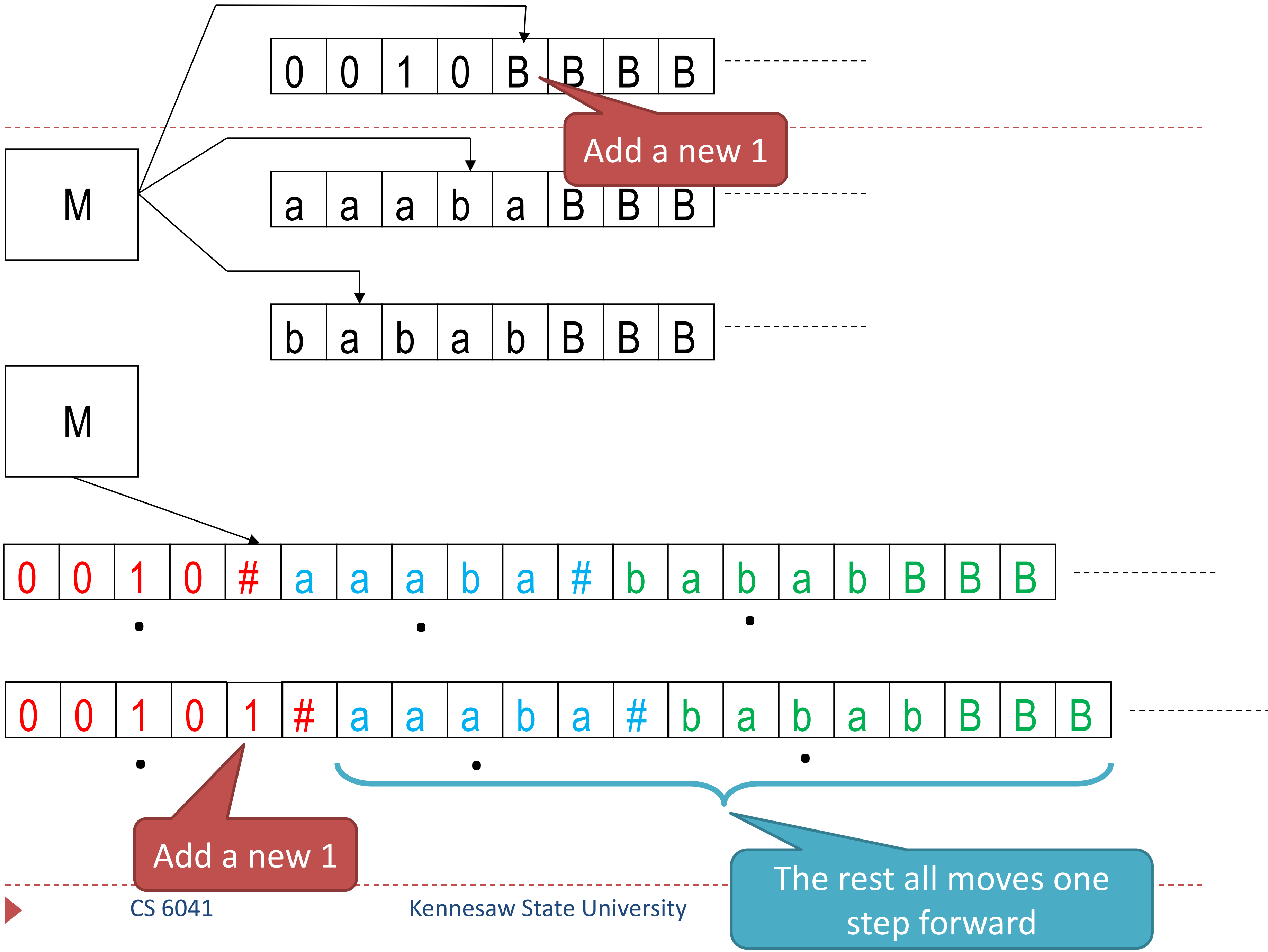
Multitape TM = single-tape TM

- How to store multi-tape on a single tape
- **How to simulate multi-head on a single tape**
- How to simulate one move on multi-tape on a single tape



- **How to simulate one move on multi-tape on a single tape**




0 0 1 0 B B B B
Add a new 1
M
a a a b a B B B
b a b a b B B B
M
0 0 1 0 # a a a b a # b a b a b B B B
0 0 1 0 1 # a a a b a # b a b a b B B B
Add a new 1
The rest all moves one step forward

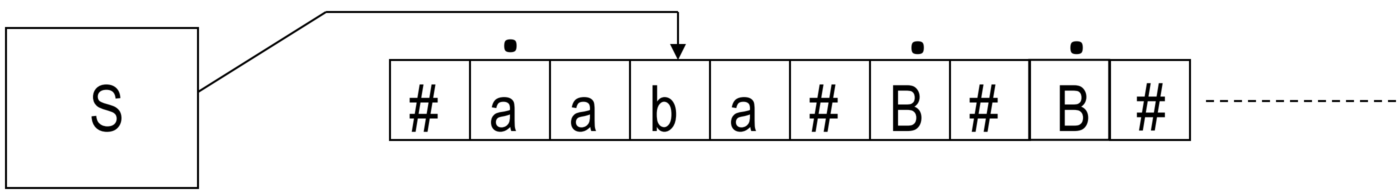
Simulate multitape TM by single-tape TM

Design single-tape TM S to simulate multitape TM M:

S="for input $w=w_1 \ldots w_n$:

1) S puts its tape in format $\#w_1w_2 \ldots w_n\#B\#B\# \ldots \#$.

2) To simulate a single move, S scans its tape from the first # to the (k + 1) # in order to determine the symbols under the virtual heads. Then S makes a second pass to update the tapes according to the way that M's transition function dictates.



3) If at any point S moves one of the virtual heads to the right onto a #, S writes a blank symbol on this tape cell and shifts the tape contents, from this cell until the rightmost #, one unit to the right. Then it continues the simulation as before.

Revisit Definition of Turing Machine

- TM $M=(Q,\Sigma,\Gamma,\delta,q_0,q_{acc},q_{rej})$

1) Q is the set of states
2) Σ is the input alphabet, not containing blank symbol $B \notin \Sigma$
3) Γ is the tape alphabet, $\Sigma \cup \{B\} \subseteq \Gamma$,
4) $\boxed{\delta: Q \times \Gamma \rightarrow Q \times \Gamma \times \{L,R\}}$ is the transition function
5) $q_0 \in Q$ is the start state
6) $q_{acc} \in Q$ is the accept state
7) $q_{rej} \in Q$ is the reject state, $q_{acc} \neq q_{rej}$

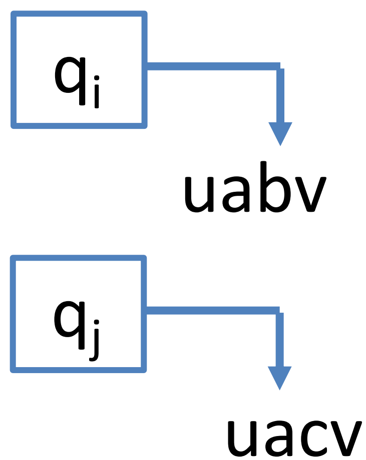
Destination is only one!

Nondeterministic Turing machine

- Nondeterministic Turing machine (NTM):
  - $\delta: Q \times \Gamma \rightarrow P(Q \times \Gamma \times \{L,R\})$

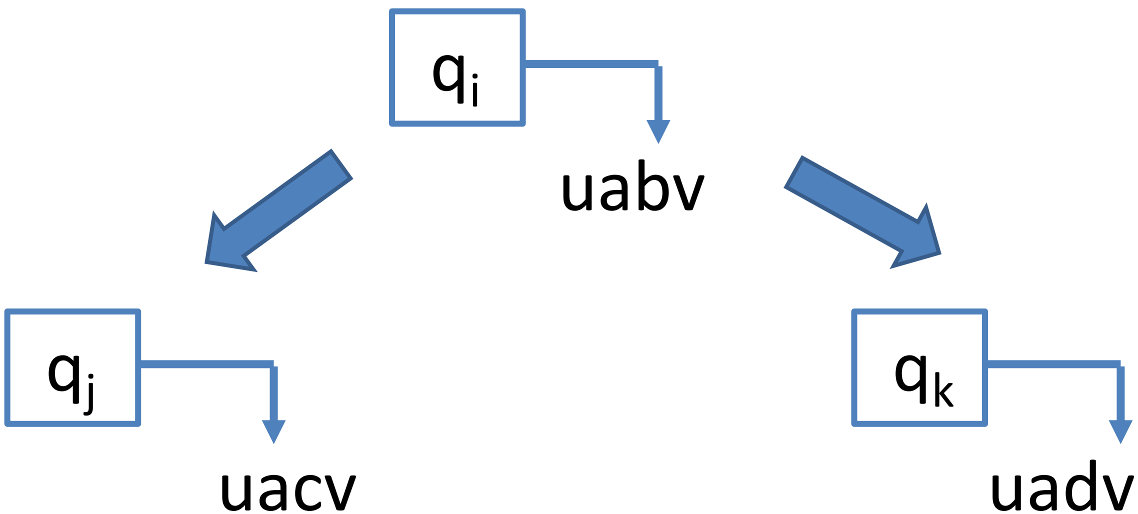
A set of {state, input, head move}

- Configuration on DTM:
  - $\boldsymbol{\delta(q_i,b)=(q_j,c,L)}$
  - $\boldsymbol{uaq_ibv \rightarrow uq_jacv}$



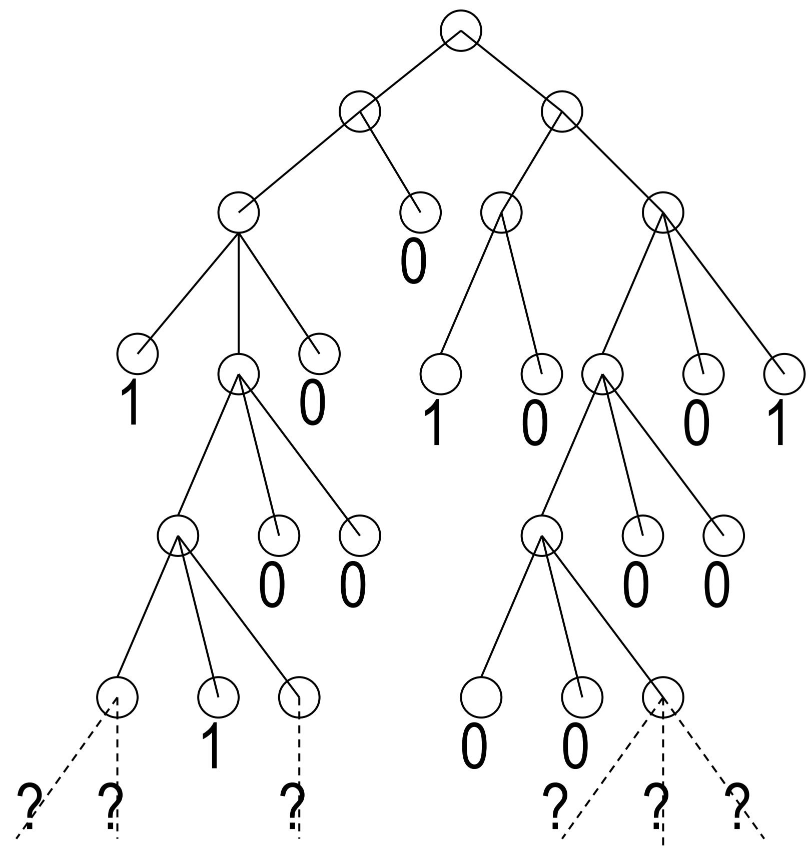
Nondeterministic Turing machine

- Configuration on NTM:
  - $\delta(q_i,b)=(q_j,c,L)$ or (q_k,d,R)
  - $uaq_ibv \rightarrow uq_jacv$ $uaq_ibv \rightarrow uadq_kv$



Nondeterministic Turing machine

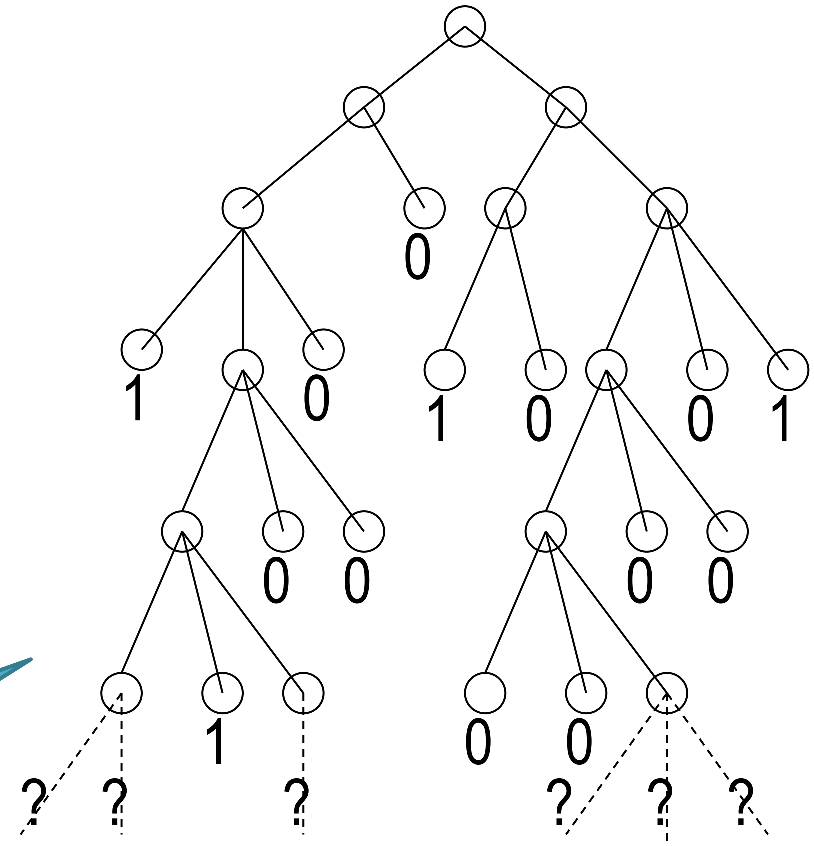
- Computation tree:
  - Nondeterministic tree



- Output:
  - Accept, if one branch accepts
  - Reject, if all branched reject
  - Loop, computation continues but accept/reject never happened

Nondeterministic TM = Deterministic TM

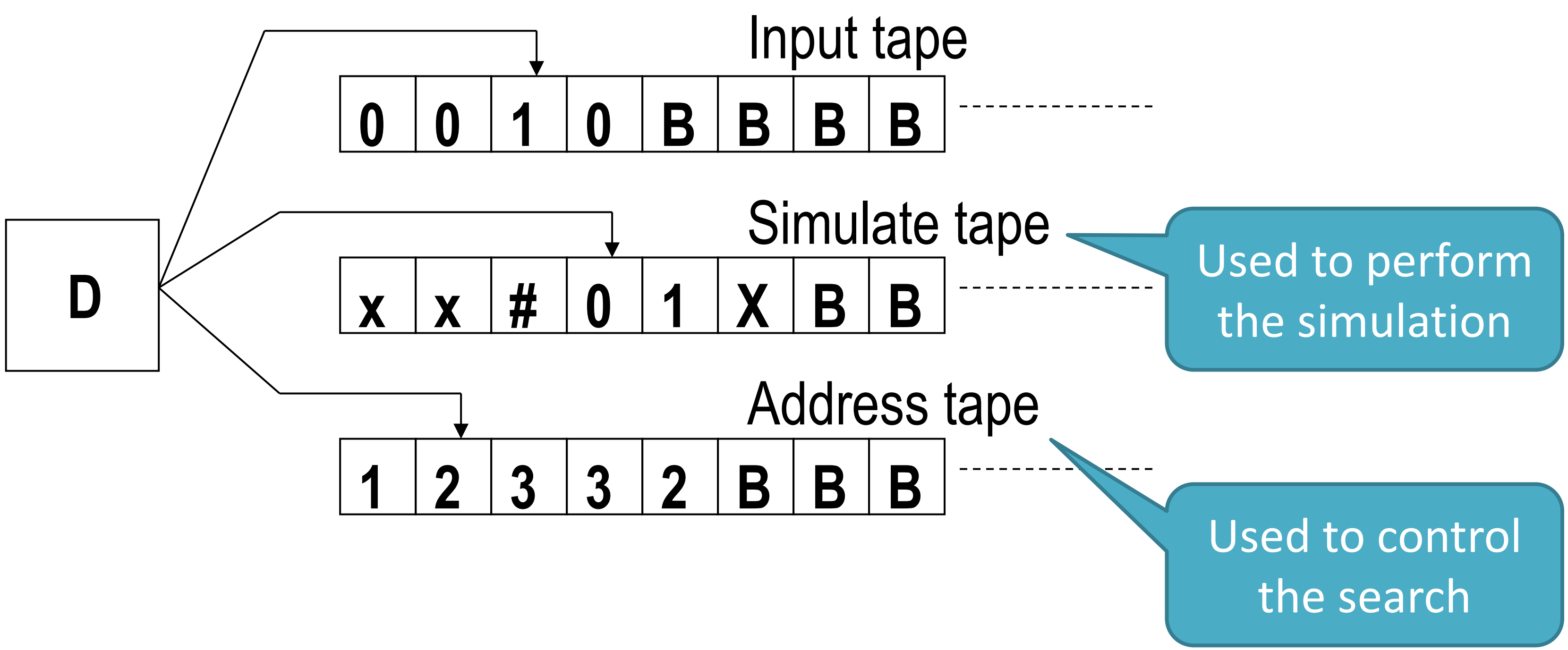
- Theorem: Every NTM has an equivalent DTM.
- Proof: use DTM D to simulate NTM N

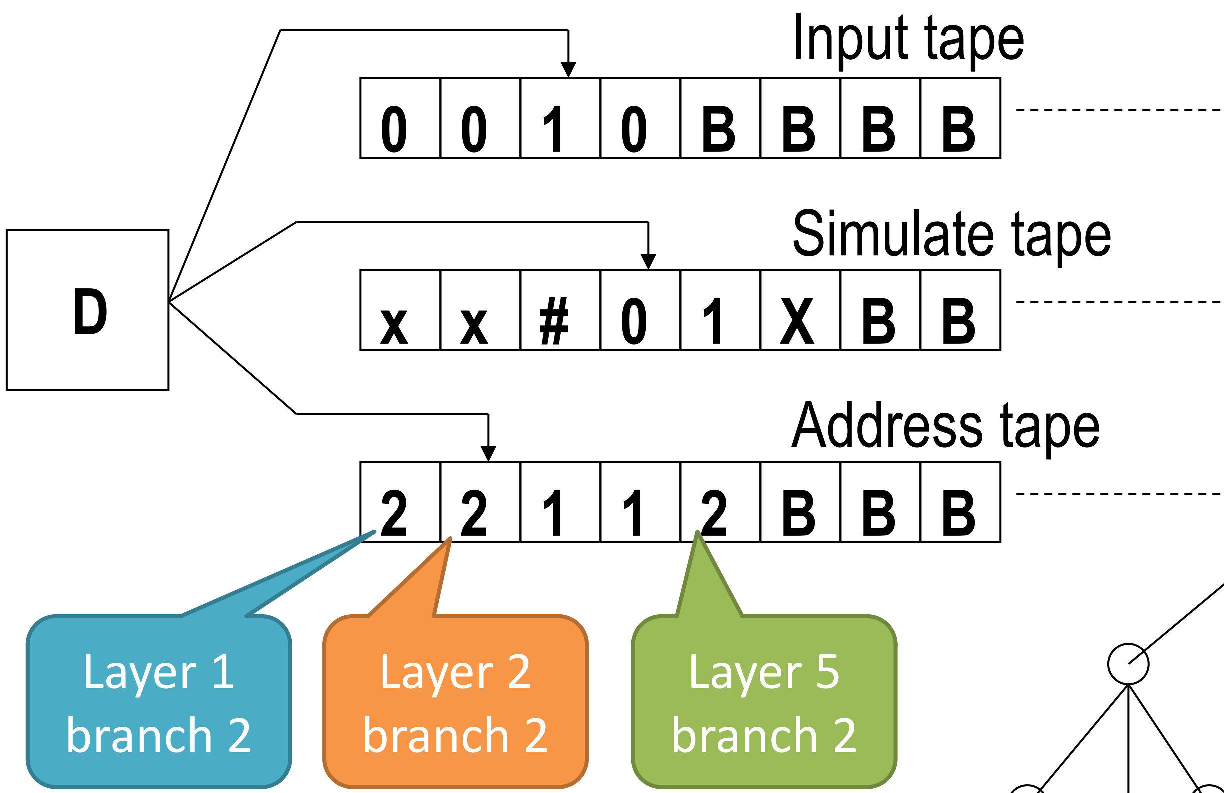


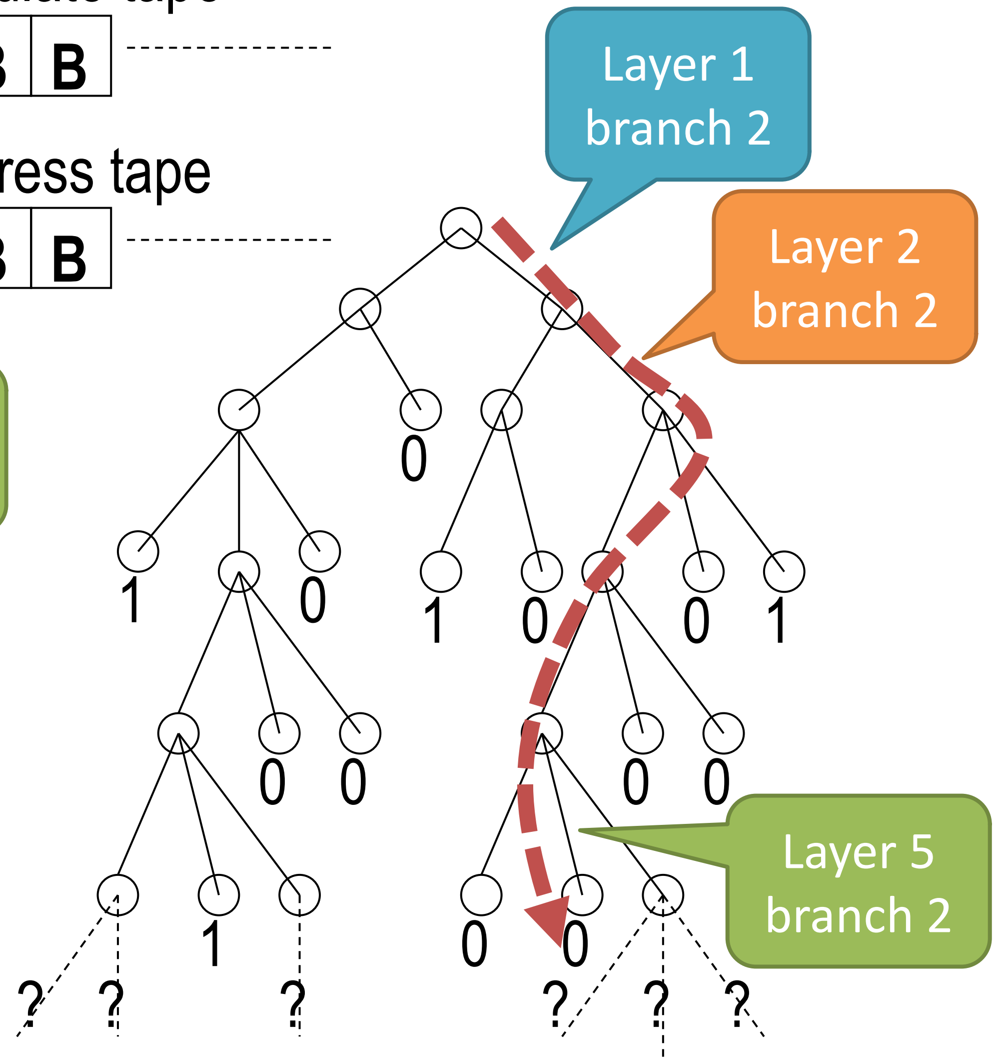
Idea: Search the tree looking for ACCEPT

Nondeterministic TM = Deterministic TM

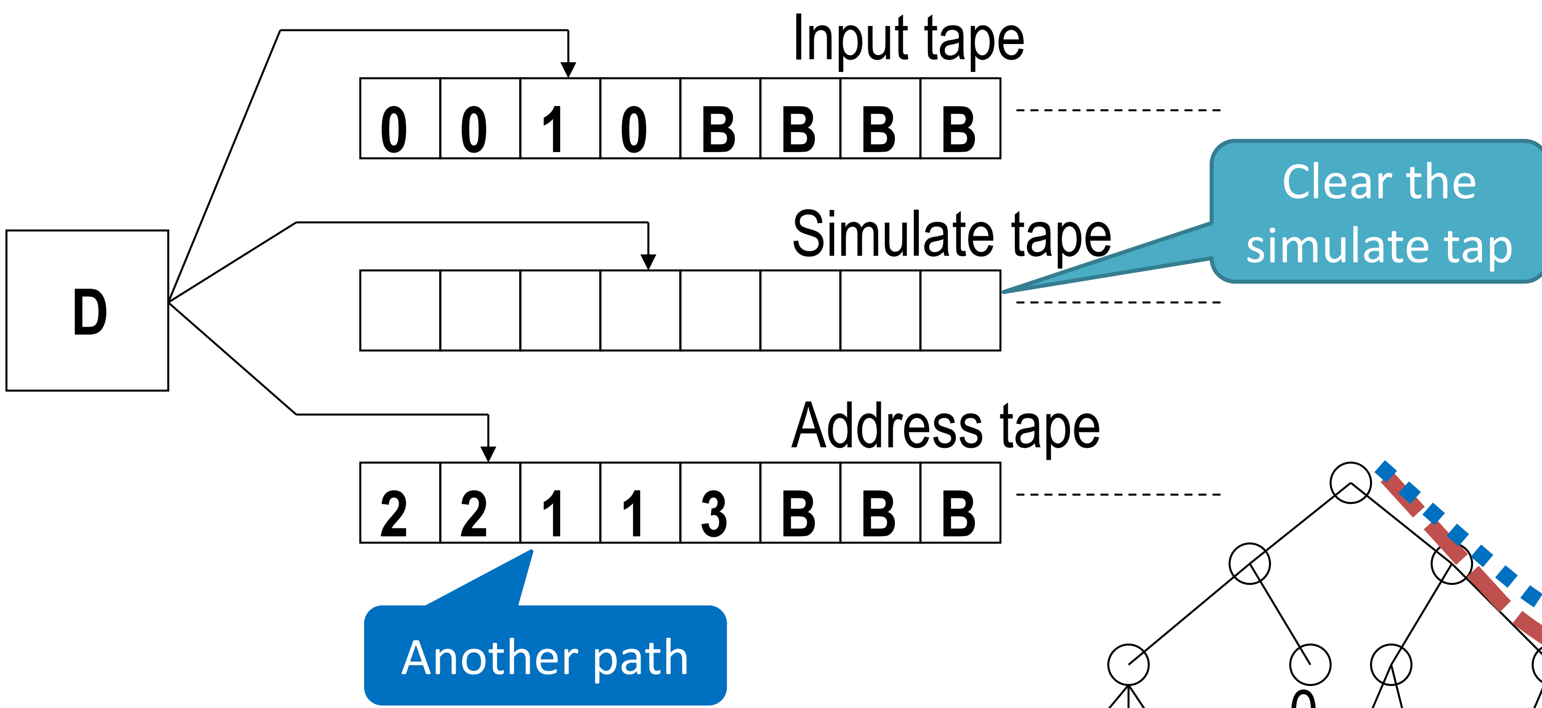
- Proof: use DTM D to simulate NTM N

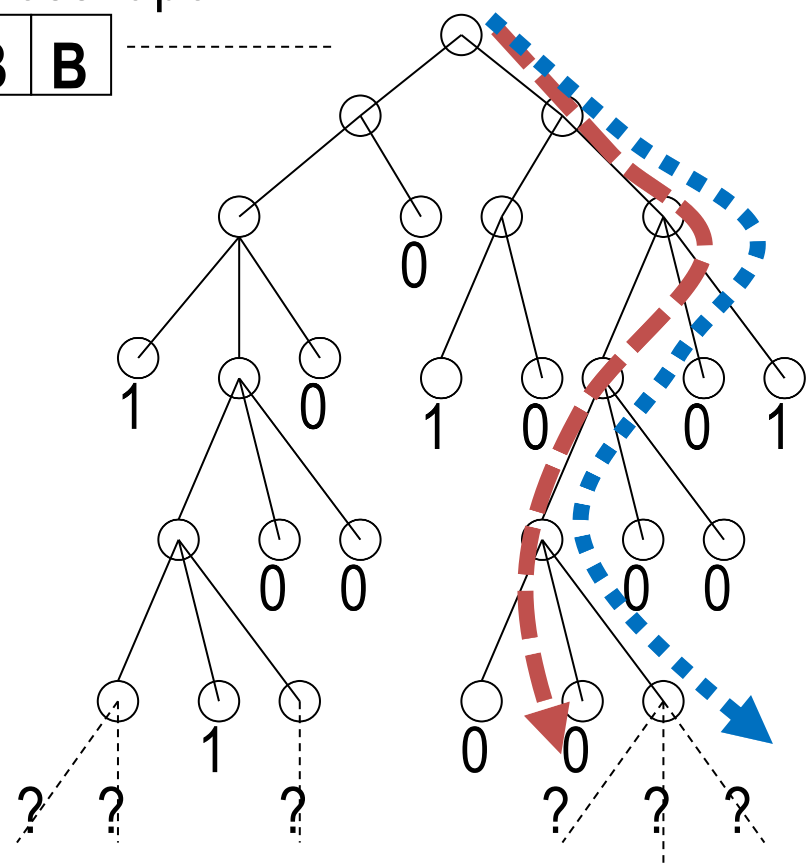






- For a given path, it is ***deterministic***. So we can use the simulate tape to test the input is accepted or not





- After simulation, clear the simulate tape and try ***another path***

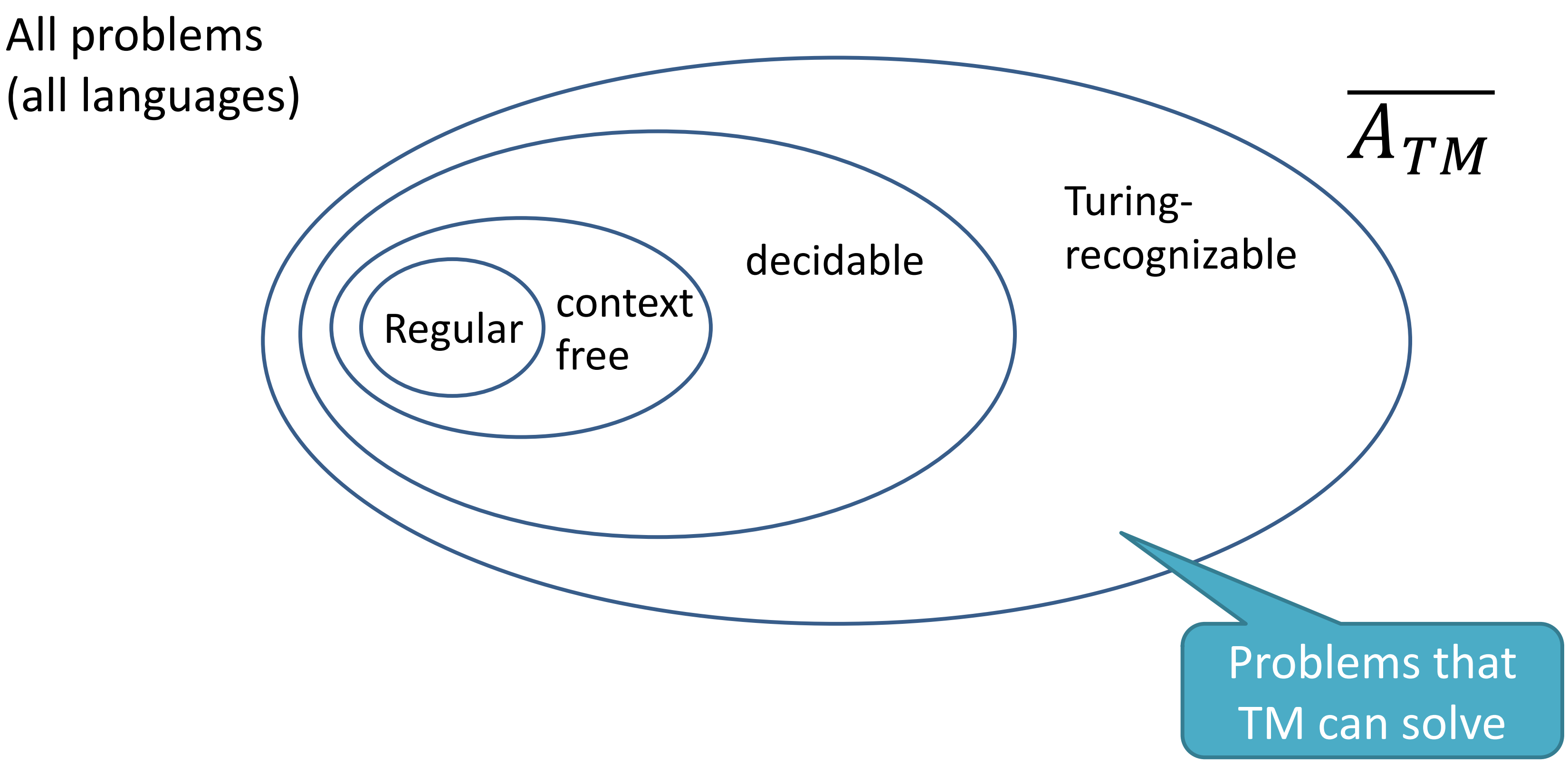
Question

- A language is Turing-recognizable iff some NTM recognizes it
  - True

- A language is decidable iff some NTM decides it.
  - True

What tasks TM can do?

- Turing machine and algorithms are equivalent in power



Conclusion

- Turing-recognizable and Turing-decidable
- Example of Turing machines
  - $\{0^{2^n}\}$
  - {w#w}
- Variants of TMs
  - Multi-tape TM
  - Nondeterministic TM