# Lecture 23 – Extensions of Turing Machines

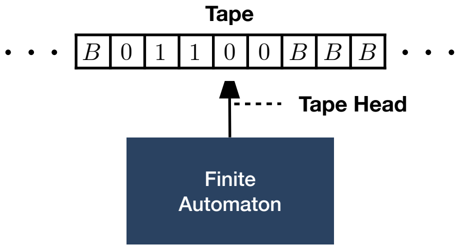
COSE215: Theory of Computation

Jihyeok Park

PLRG

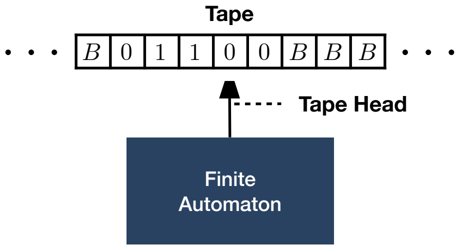
2023 Spring

# Recall



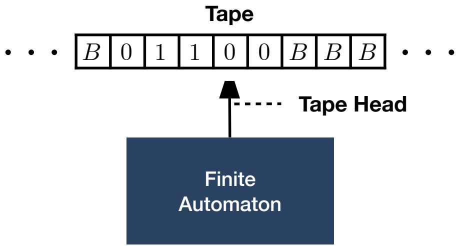

- A **Turing machine (TM)** is a finite automaton with a **tape**.
- A language accepted by a TM is **Recursively Enumerable**.

# Recall




- A **Turing machine (TM)** is a finite automaton with a **tape**.
- A language accepted by a TM is **Recursively Enumerable**.
- What happens if we define **other extensions** of TMs?
- Are they **more powerful** than TMs?

# Recall

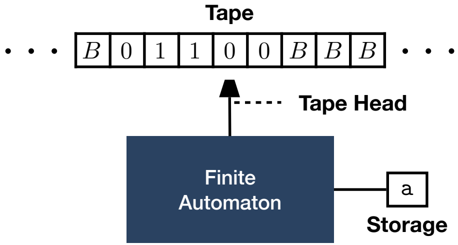



- A **Turing machine (TM)** is a finite automaton with a **tape**.
- A language accepted by a TM is **Recursively Enumerable**.
- What happens if we define **other extensions** of TMs?
- Are they **more powerful** than TMs? **NO!!**

# Contents

# TMs with Storage

We can define a TM with a **storage**:



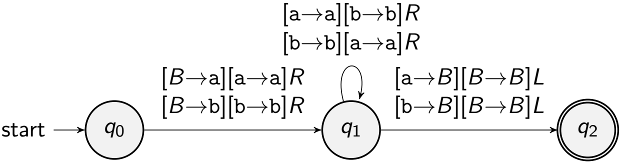

It has additional **storage** affecting the transition function:

$$\delta : Q \times \Gamma \times \Gamma \rightharpoonup Q \times \Gamma \times \Gamma \times \{L, R\}$$

# TMs with Storage – Example

$$L(M) = \{\texttt{ab}^n \text{ or } \texttt{ba}^n \mid n \geq 0\}$$

The following TM with storage accepts $L(M)$, and see the example for $\texttt{abb} \in L(M)$.[1]




[1] https://plrg.korea.ac.kr/courses/cose215/materials/tm-storage-abn-or-ban.pdf

# TMs with Storage are Equivalent to TMs

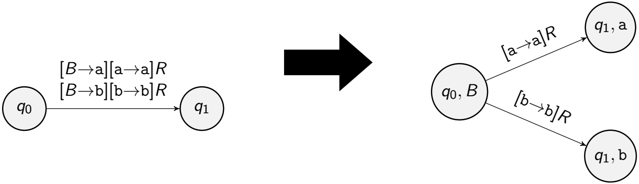
**Theorem**

*A language accepted by a* **TM with storage** *is* **recursively enumerable** *(i.e., accepted by a standard* **TM***).*

**Proof)** We can define an equivalent standard TM by using pairs of states and symbols in the storage as its states:

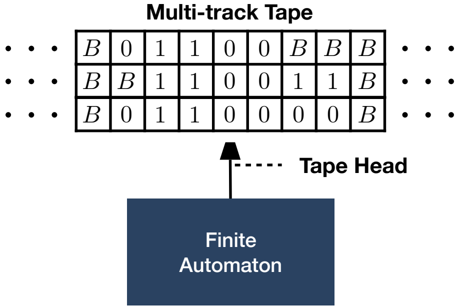
$$\delta'((q, a), b) = \delta(q, a, b)$$

where $Q' = Q \times \Gamma$ and $\delta' : Q' \times \Gamma \rightharpoonup Q' \times \Gamma \times \{L, R\}$.
For example,



# Multi-track TMs

We can define a TM with a **multi-track tape**:



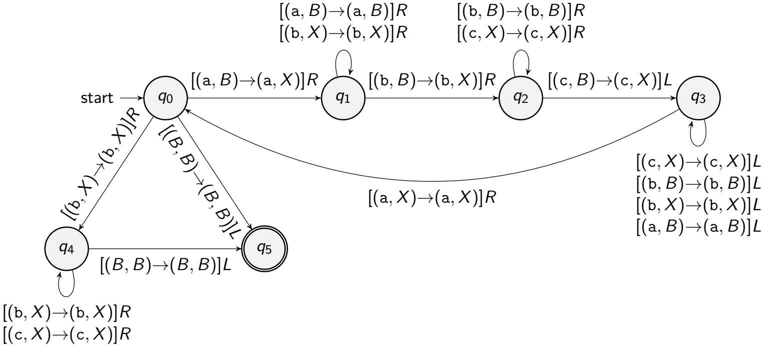

It has a tape with $n$ **tracks** and a **single tape head**:

$$\delta : Q \times \Gamma^n \rightharpoonup Q \times \Gamma^n \times \{L, R\}$$

# Multi-track TMs – Example

$$L(M) = \{\mathtt{ab}^n \text{ or } \mathtt{ba}^n \mid n \geq 0\}$$

The following multi-track TM accepts $L(M)$, and see the example for $\mathtt{aabbcc} \in L(M)$.[2]




2 https://plrg.korea.ac.kr/courses/cose215/materials/tm-multi-track-an-bn-cn.pdf

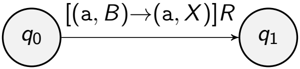
**Theorem**

*A language accepted by a* **multi-track TM** *is* **recursively enumerable** *(i.e., accepted by a standard* **TM***).*

**Proof)** We can define an equivalent standard TM by using $n$-tuples of symbols as a single symbol:

$$\delta'(q, \alpha) = \delta(q, \alpha)$$

where $\Gamma' = \Gamma^n$ and $\delta' : Q \times \Gamma' \rightharpoonup Q \times \Gamma' \times \{L, R\}$.
For example,

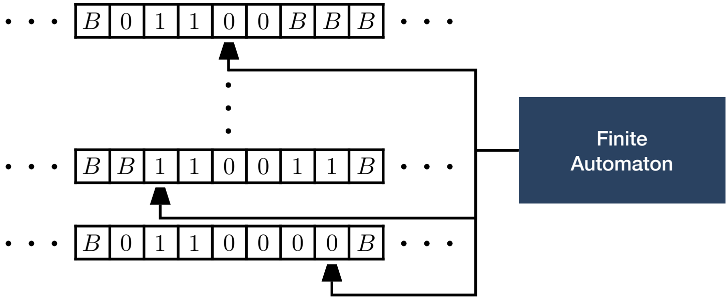



| | | | | | |
|---|---|---|---|---|---|
| ⋯ | B | a | b | B | ⋯ |
| ⋯ | B | X | B | B | ⋯ |

| | | | | | |
|---|---|---|---|---|---|
| ⋯ | (B, B) | (a, X) | (b, B) | (B, B) | ⋯ |

# Multi-tape TMs

We can define a TM with **multiple tapes**:



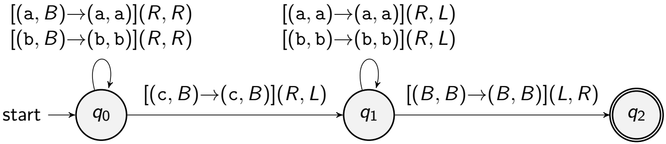

It has $n$ **tapes**, and each tape has its **own head** that can move independently:

$$\delta : Q \times \Gamma^n \rightharpoonup Q \times (\Gamma \times \{L, R\})^n$$

# Multi-tape TMs – Example

$$L(M) = \{w\mathtt{c}w^R \mid w \in \{\mathtt{a}, \mathtt{b}\}^*\}$$

The following multi-tape TM accepts $L(M)$, and see the example for $\mathtt{abbcbba} \in L(M)$.[3]




[3] https://plrg.korea.ac.kr/courses/cose215/materials/tm-multi-tape-w-c-wr.pdf

# Multi-tape TMs are Equivalent to Standard TMs

## Theorem

*A language accepted by a* **multi-tape TM** *is* **recursively enumerable** *(i.e., accepted by a standard* **TM***).*

**Proof)** For a given $n$-tape TM, we can define an equivalent $2n$-track TM with storage by using **odd** tracks for the original **tapes** and **even** tracks for the **tape heads**:

| | | | | | | |
|---|---|---|---|---|---|---|
| $\cdots$ | B | a | b | b | B | $\cdots$ |
| | | $\uparrow$ | | | | |
| $\cdots$ | B | b | c | a | B | $\cdots$ |
| | | | | $\uparrow$ | | |

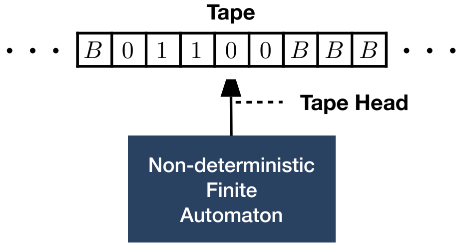
$\Rightarrow$

| | | | | | | |
|---|---|---|---|---|---|---|
| $\cdots$ | B | a | b | b | B | $\cdots$ |
| $\cdots$ | B | X | B | B | B | $\cdots$ |
| $\cdots$ | B | b | c | a | B | $\cdots$ |
| $\cdots$ | B | B | B | X | B | $\cdots$ |

We can simulate one-step in the $n$-tape TM by gathering all the symbols pointed by the $n$ heads into the storage, and then taking the action same as done by the $n$-tape TM.

# Non-deterministic TMs

We can define a TM with **non-deterministic transitions**:

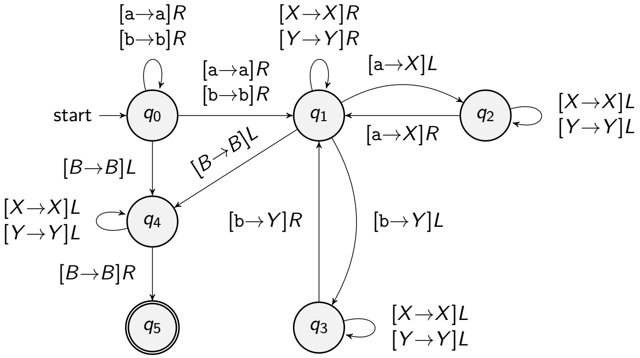



It has a **non-deterministic transition function**:

$$\delta : Q \times \Gamma \rightarrow \mathcal{P}(Q \times \Gamma \times \{L, R\})$$

$$L(M) = \{ww^R \mid w \in \{\mathtt{a}, \mathtt{b}\}^*\}$$

The following multi-tape TM accepts $L(M)$, and see the example for abba $\in L(M)$.[4]

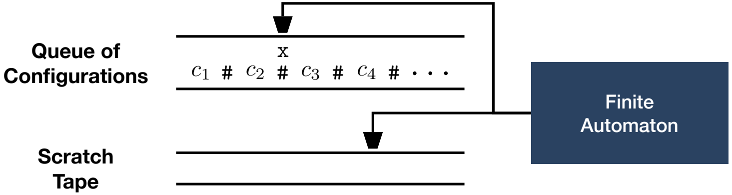



[4] https://plrg.korea.ac.kr/courses/cose215/materials/ntm-w-wr.pdf

**Theorem**

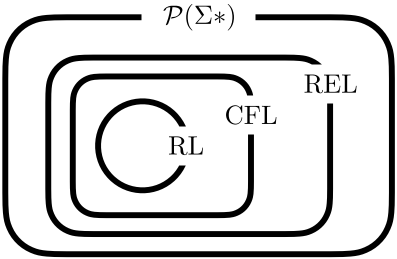
*A language accepted by a* **non-deterministic TM** *is* **recursively enumerable** *(i.e., accepted by a standard* **TM***).*

**Proof)** For a given non-deterministic TM, we can define an equivalent 2-tape TM: 1) a 2-track tape to maintain a **queue of configurations** and 2) a normal track to **simulate** the tape of the original TM.



# More Extensions of TMs

- There are more extensions of TMs:
  - TMs with **Stay Option** – $L$: Left, $R$: Right, and $S$: **Stay**
  - **Queue Automata** – Automata with **Queue**
  - **Random Access Machines** – TMs with **Random Access Memory**
  - ...
- They are all **equivalent** to TMs.
- A standard **TM** is the **most powerful model of computation**.




TM = ETM

(Turing Machine) (All Extensions of TMs)

=

REL

(Recursively Enumerable Language)

# Summary

1. Extensions of Turing Machines
   - TMs with Storage
   - Multi-track TMs
   - Multi-tape TMs
   - Non-deterministic TMs
   - More Extensions of TMs

# Next Lecture

- The Origin of Computer Science

Jihyeok Park
jihyeok_park@korea.ac.kr
https://plrg.korea.ac.kr