

# Variants of Turing Machines

CSCI 3130 Formal Languages and Automata Theory

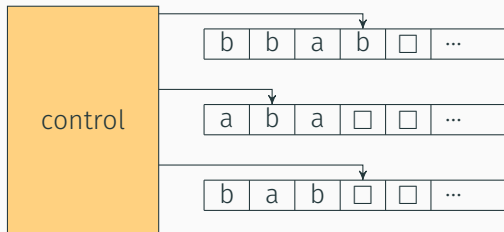
---

Siu On CHAN

Fall 2020

Chinese University of Hong Kong

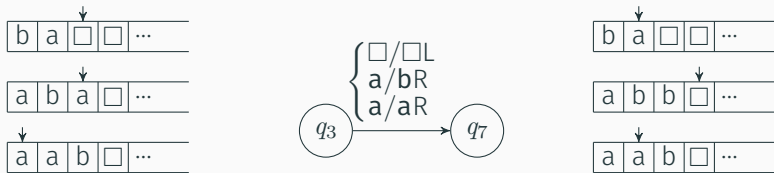
# Multitape Turing machine



Transitions may depend on the contents of all cells under the heads

Different tape heads can move independent

# Multitape Turing machine

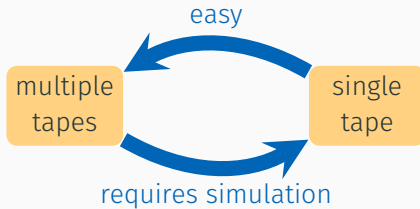


Multiple tapes are convenient

Some tapes can serve as temporary storage

# How to argue equivalence

Multitape Turing machines are **equivalent** to single-tape Turing machines



# Simulating multitape Turing machine

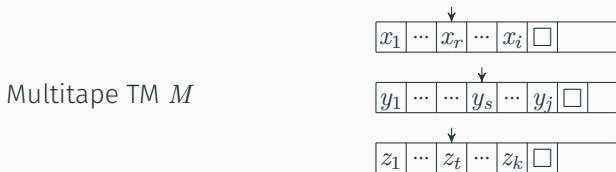


$$\Gamma = \{a, b, \square, \dot{a}, \dot{b}, \dot{\square}, \#\}$$

# Simulating multitape Turing machine

We show how to **simulate** a multitape Turing machine on a single tape Turing machine

To be specific, let's simulate a 3-tape TM

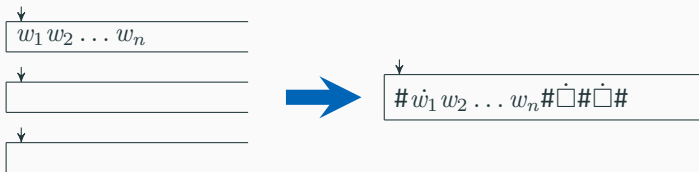


Single tape TM  $S$



# Simulating multitape Turing machine

## Single-tape TM: Initialization



$S$ : On input  $w_1 \dots w_n$ :

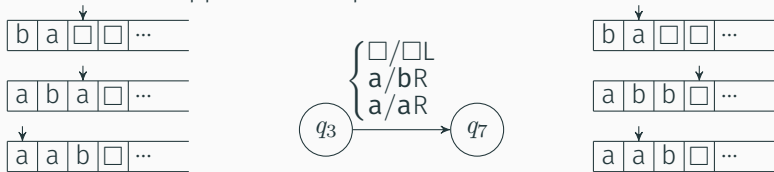
Replace tape contents by  $\# w_1 w_2 \dots w_n \# \square \# \square \#$

Remember that  $M$  is in state  $q_0$

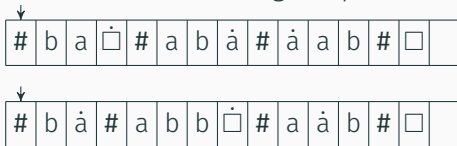
# Simulating multitape Turing machine

## Single-tape TM: Simulating multitape TM moves

Suppose Multitape TM  $M$  moves like this:



We simulate the move on single-tape TM  $S$  like this





# Simulating multitape Turing machine

$S$  given input  $w_1 \dots w_n$ :

Replace tape contents by  $\# \dot{w}_1 w_2 \dots w_n \# \square \# \square$

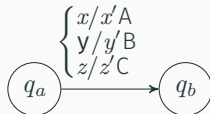
Remember (in state) that  $M$  is in state  $q_0$

$S$  simulates a step of  $M$ :

Make a pass over tape to find  $\dot{x}, \dot{y}, \dot{z}$

$\downarrow$   
 $\# x_1 x_2 \dots \dot{x} \dots x_i \# y_1 y_2 \dots \dot{y} \dots y_j \# z_1 z_2 \dots \dot{z} \dots z_k \#$

If  $M$  at state  $q_a$  has transition



update state/tape accordingly

If  $M$  reaches accept (reject) state,  $S$  accepts (rejects)

To **simulate** a model  $M$  by another model  $N$ :

Say how the state and storage of  $N$  is used to represent the state and storage of  $M$

Say what should be initially done to convert the input of  $N$

Say how each transition of  $M$  can be implemented by a sequence of transitions of  $N$