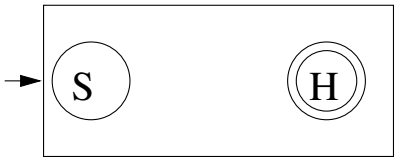
Section: Turing Machines - Building Blocks

1. Given Turing Machines M1 and M2

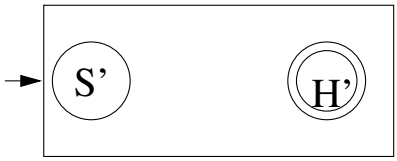
Notation for

- **Run M1**
- **Run M2**

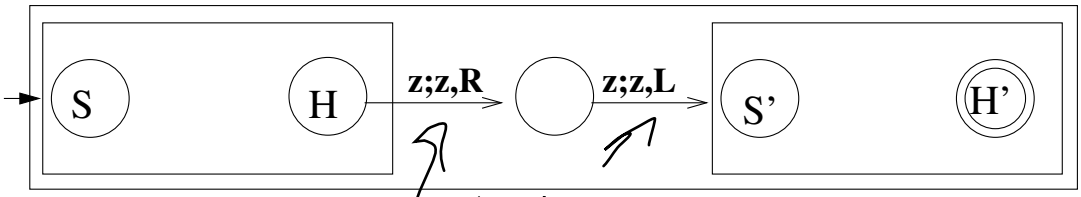
M1



M2



→ M1 → M2

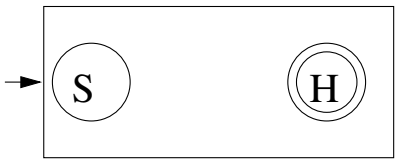


lots of arcs

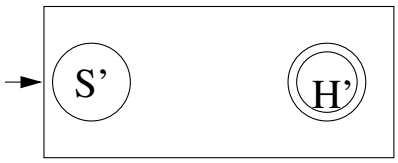
z represents any symbol in

2. Given Turing Machines M1 and M2

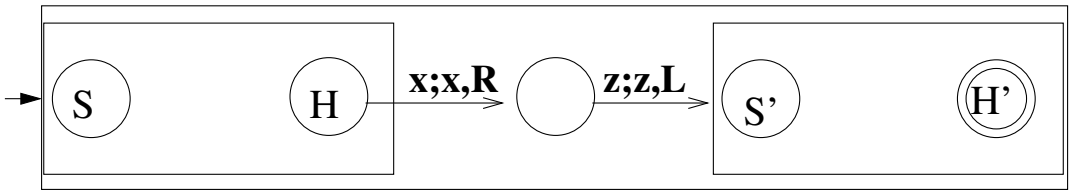
M1



M2



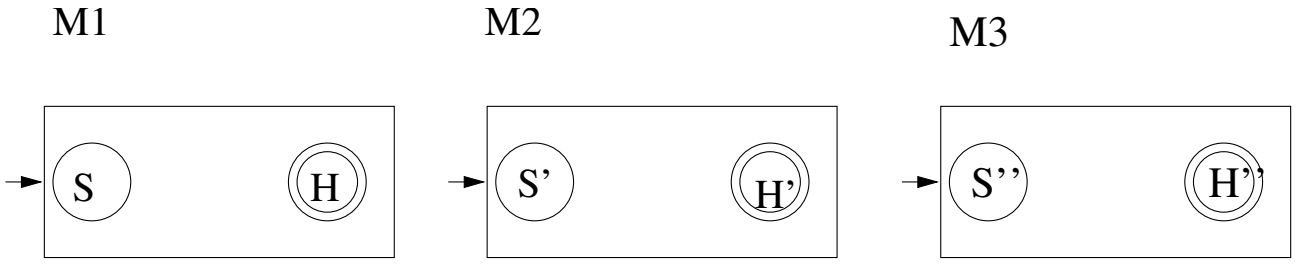
$\rightarrow$ M1 $\overset{x}{\rightarrow}$ M2

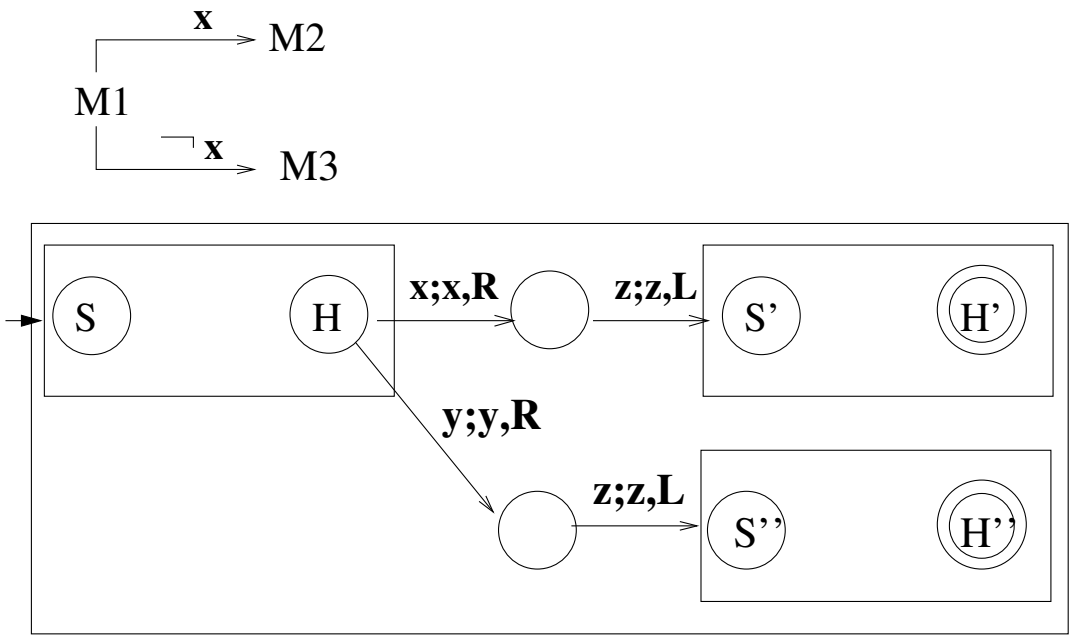


z represents any symbol in

x is an element of

3. Given Turing Machines M1, M2, and M3





x is an element of ⌜

y is any element except x from ⌜

z is any element from ⌜

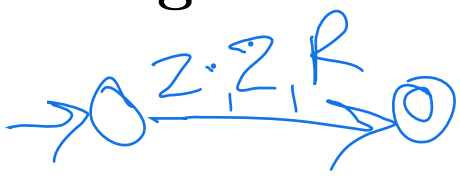
More Notation for Simplifying Turing Machines

Suppose Γ={a,b,c,B}

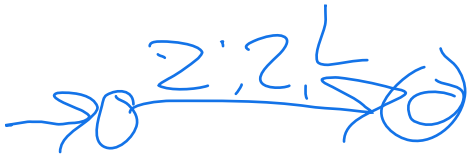
z is any symbol in Γ

x is a specific symbol from Γ

1. **s - start**
2. **R - move right**

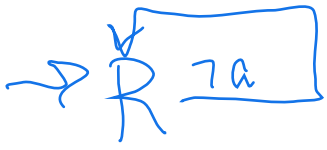


3. **L - move left**



4. **x - write x (and don't move)**

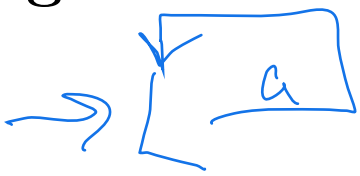
5. **R$_a$ - move right until you see an a**



6. **L_a - move left until you see an a**

7. **$R_{\neg a}$ - move right until you see anything that is not an a**

8. **$L_{\neg a}$ - move left until you see anything that is not an a**



9. **h - halt in a final state**
10. $\overset{a,b}{\rightarrow}\}\ \overset{w}{\rightarrow}$

    If the current symbol is a or b, let w represent the current symbol.

Example

Assume input string $w \in \Sigma^+$, $\Sigma = \{a, b\}$.

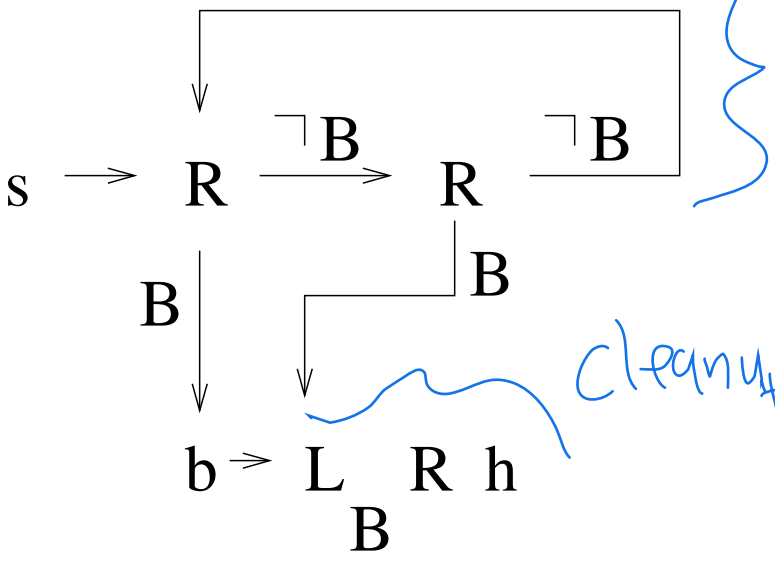
If $|w|$ is odd, then write a b at the end of the string. The tape head should finish pointing at the leftmost symbol of w.

input: bab, output: babb

input: ba, output: ba

assuming $|x| \geq 1$

bab



toggling to determine even or odd length

cleanup

What is the running time? $|w| = n$

$\Theta(n)$

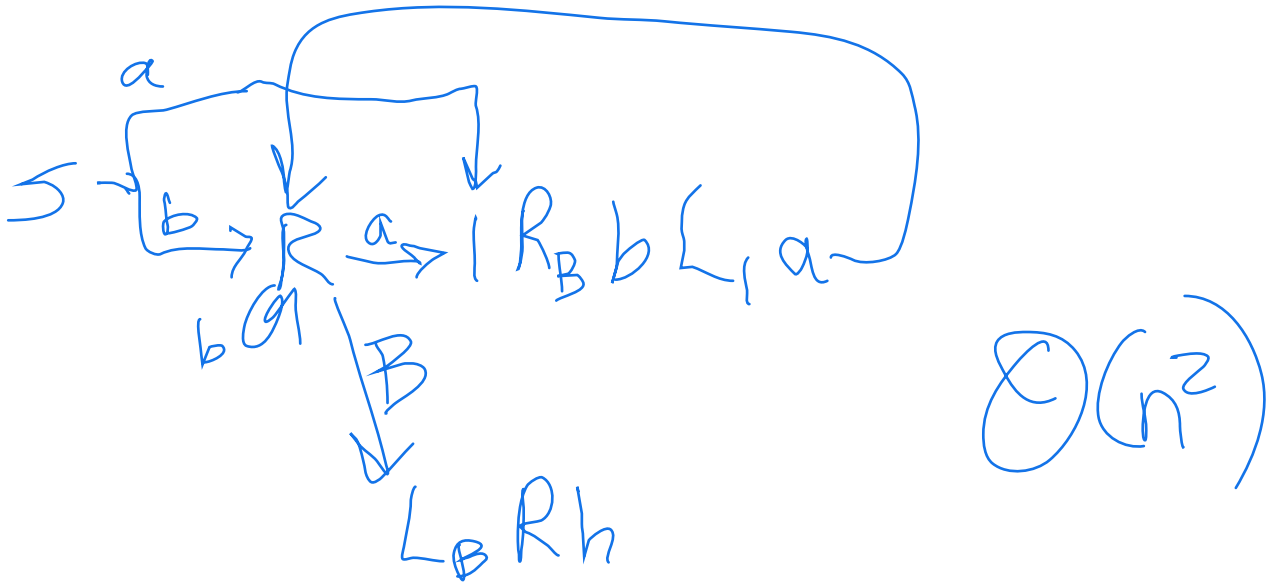
Example

Assume input string $w \in \Sigma^+$, $\Sigma = \{a, b\}$, $|w| > 0$

For each a in the string, append a b to the end of the string.

input: $abbabb$, output: $abbabbbb$

The tape head should finish pointing at the leftmost symbol of w.



Turing's Thesis Any computation that can be carried out by a mechanical means can be performed by a TM.

Definition: An *algorithm* for a function f:D→R is a TM M, which given input d∈D, halts with answer f(d)∈R.

Example: $f(x+y) = x+y$**,** x **and** y **unary numbers.**

|x|≥1 , |y|≥1

start with: **111+1111**

↑

end with: **1111111**

↑

O(m)

sR_+1L_BRBRh

|x|=m
|y|=n

$sR_+1R_BLBL_BRh$

O(m+n)

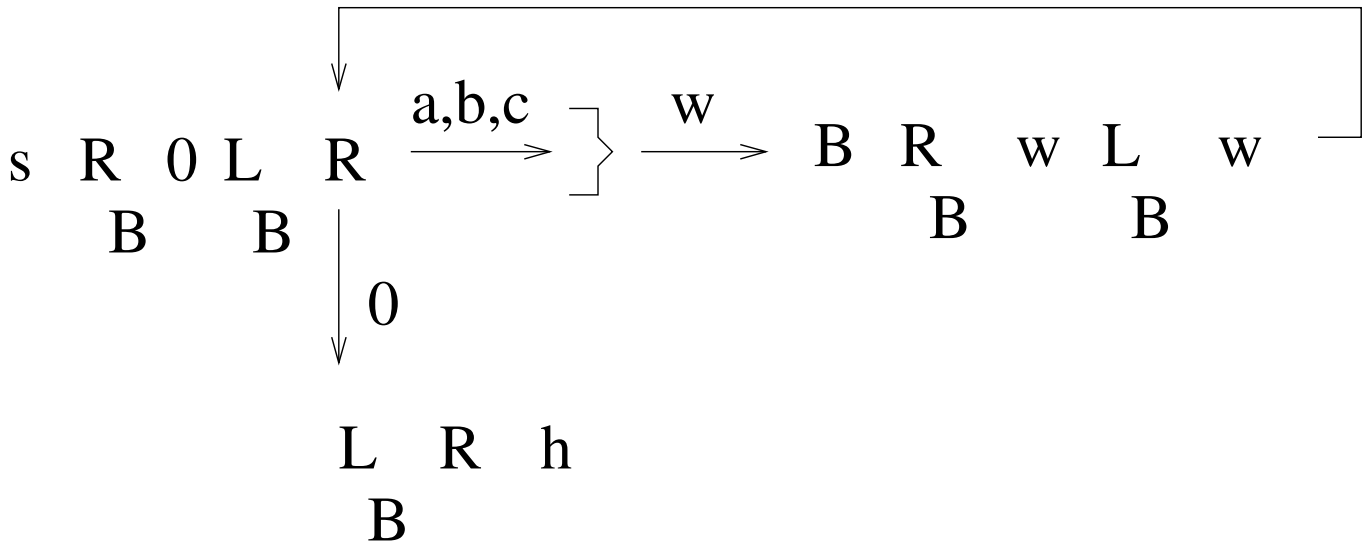
Example: Copy a String, f(w)=w0w, w$\in \Sigma^*$, $\Sigma = \{a, b, c\}$

Denoted by C

start with:	**abac**
	↑
end with:	**abac0abac**
	↑

Algorithm:

- **Write a 0 at end of string**
- **For each symbol in string**
  - **make a copy of the symbol**



start with abc

a B abc0abc

$\Theta(n^2)$

Example: Shift the string that is to the left of the tape head to the right,

denoted by S_R (shift right)

Below, "ba" is to the left of the tape head, so shift "ba" to the right.

start with: aaBbabca

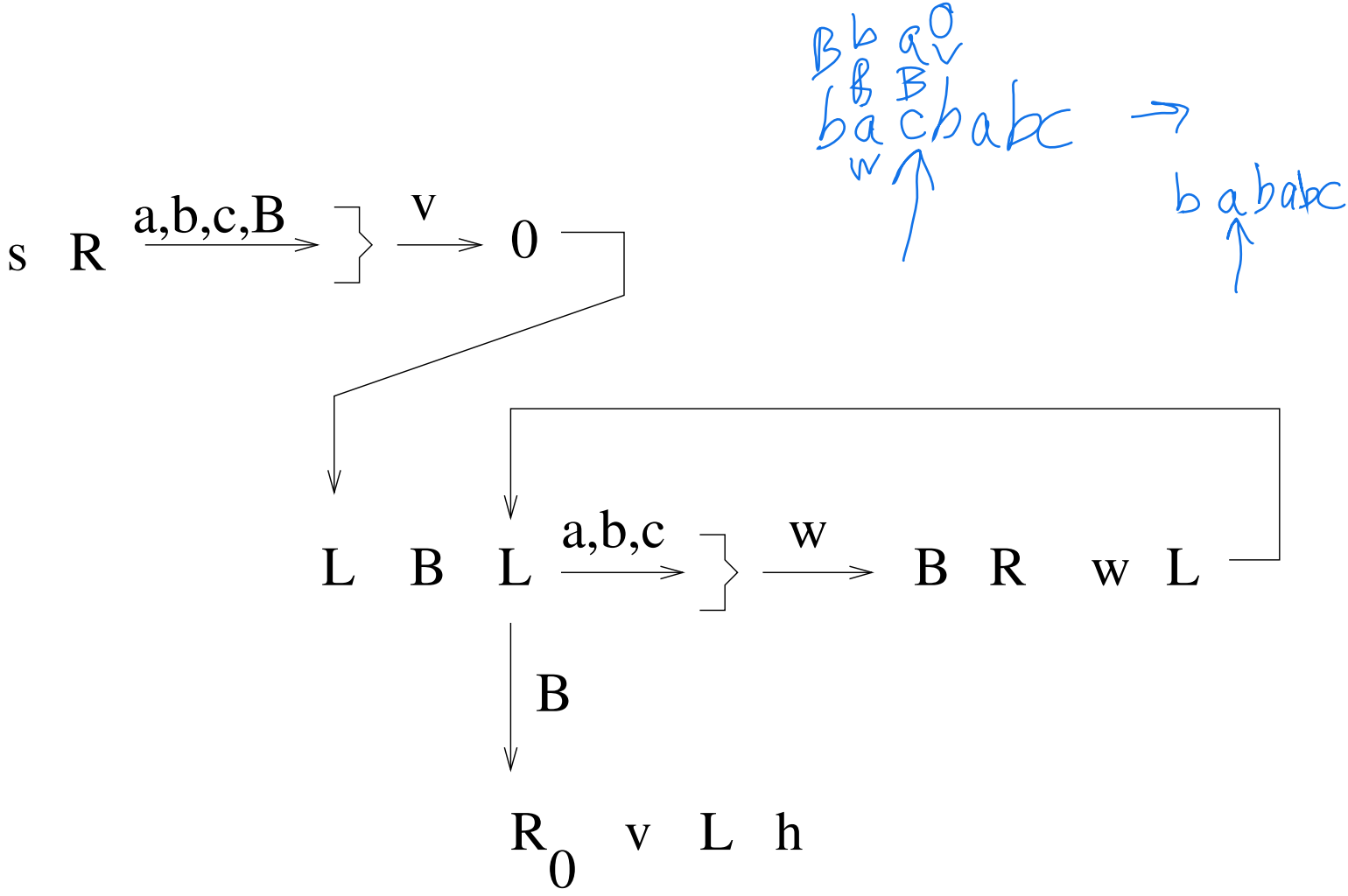
↑

end with: aaBBbaca

↑

Algorithm:

- **remember symbol to the right and erase it**
- **for each symbol to the left do**
  - **shift the symbol one cell to the right**
- **replace first symbol erased**
- **move tape head to appropriate position**


s R
a,b,c,B
v
0
L B L
a,b,c
w
B R w L
B
R0 v L h

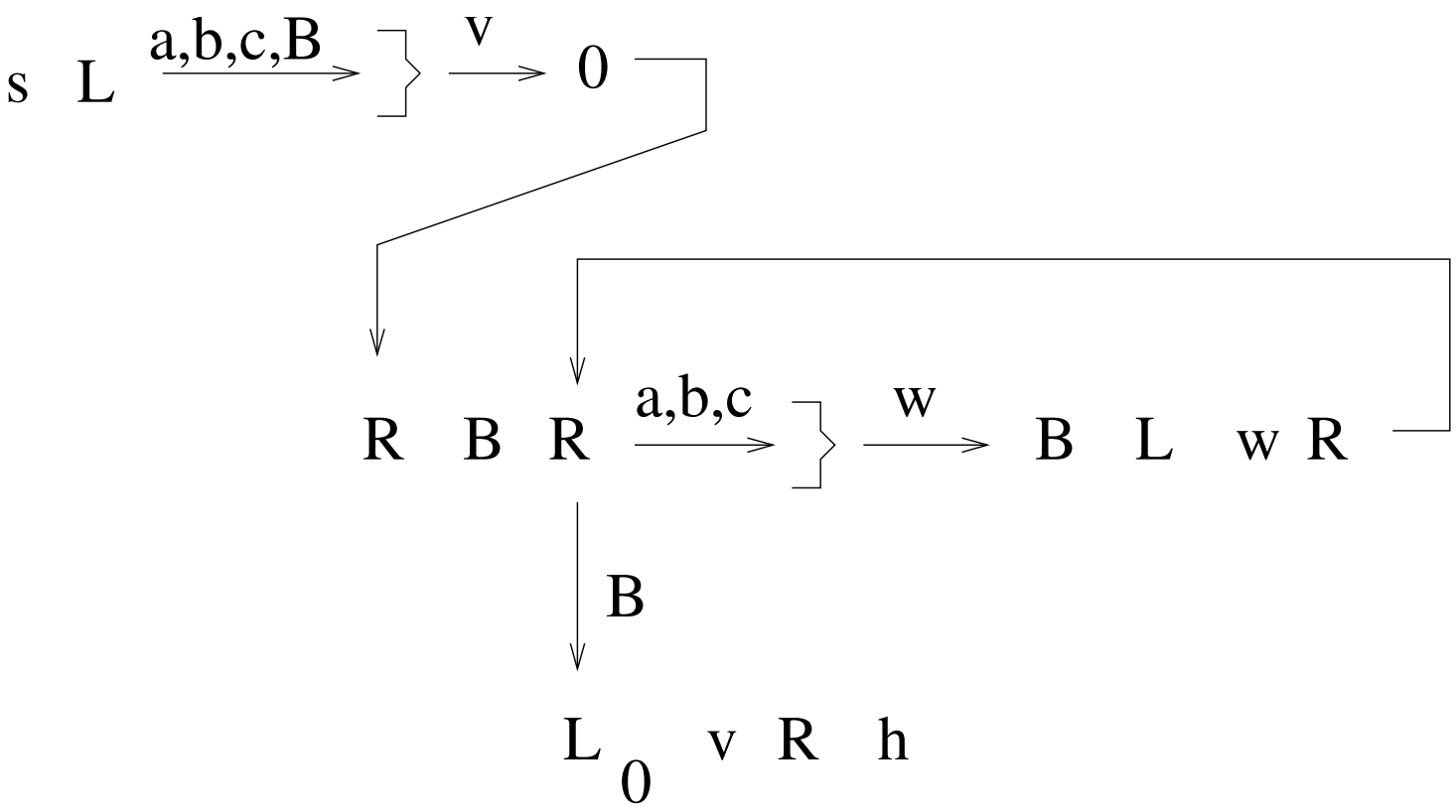
Example: Shift the string that is to the right of tape head to the left,

denote by $\mathbf{S}_L$ (shift left)

start with: **babcaBba**

end with: **bacaBBba**

(similar to $\mathbf{S}_R$)


s L
a,b,c,B
v
0
R B R
a,b,c
w
B L w R
B
L 0 v R h

Example: Add unary numbers

This time use shift.

Example: Multiply two unary numbers, f(x∗y)=x∗y, x and y unary numbers. Assume x,y>0.

start with:	**1111∗11**
	↑
end with:	**11111111**
	↑