Combining Turing Machines

We will define notation that will make it easier to look at more complicated Turing machines

1. Given Turing Machines M1 and M2
   Notation for
   - Run M1
   - Run M2

2. Given Turing Machines M1 and M2
   Notation for
   - Run M1
   - If x is current symbol
     - then Run M2
3. Given Turing Machines M1, M2, and M3

Notation for

- Run M1
- If x is current symbol
  - then Run M2
  - else Run M3

More Notation for Simplifying Turing Machines

Suppose \( \Gamma = \{a,b,c,B\} \)

- \( z \) is any symbol in \( \Gamma \)
- \( x \) is a specific symbol from \( \Gamma \)

1. \( s \) - start
2. \( R \) - move right
3. L - move left

4. x - write x (and don’t move)

5. R\textsubscript{a} - move right until you see an a

6. L\textsubscript{a} - move left until you see an a

7. R\textsubscript{¬a} - move right until you see anything that is not an a

8. L\textsubscript{¬a} - move left until you see anything that is not an a

9. h - halt in a final state

10. \begin{align*}
    & a, b \\
    \rightarrow & \ w
\end{align*}

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

What is the running time?
**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: \( abbabbb \)

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 \( f:D \rightarrow R \) is a TM M, which given input \( d \in D \), halts with answer \( f(d) \in R \).

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

\[
\begin{array}{c|c}
\text{start with:} & 111+1111 \\
\uparrow & \\
\text{end with:} & 1111111 \\
\uparrow & \\
\end{array}
\]
Example: Copy a String, \( f(w) = w0w \), \( w \in \Sigma^* \), \( \Sigma = \{a, b, c\} \)

Denoted by \( C \)

\[
\begin{align*}
\text{start with:} & \quad \text{abac} \\
& \uparrow \\
\text{end with:} & \quad \text{abac0abac} \\
& \uparrow
\end{align*}
\]

Algorithm:

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

\[
\begin{array}{c}
\text{s R 0 L } \\
\text{B B}
\end{array}
\xrightarrow{a,b,c}
\begin{array}{c}
\text{w R B} \\
\text{w L B}
\end{array}
\xrightarrow{w}
\begin{array}{c}
\text{B R w L w} \\
\text{B B}
\end{array}
\]

\[
\begin{array}{c}
\text{L R h} \\
\text{B}
\end{array}
\]
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.

\[
\begin{align*}
\text{start with:} & \quad \text{aaBbabca} \\
\uparrow & \\
\text{end with:} & \quad \text{aaBBbaca} \\
\uparrow & 
\end{align*}
\]

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
Example: Shift the string that is to the right of tape head to the left, denote by $S_L$ (shift left)

\[
\begin{align*}
\text{start with:} & \quad \text{babcaBba} \\
\uparrow \quad & \quad \uparrow \\
\text{end with:} & \quad \text{bacaBBba} \\
\end{align*}
\]

(similar to $S_R$)
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
↑