Another TM Design

- We've designed a TM for \( \{0^n1^n \mid n \in \mathbb{N}\} \).
- Consider this language over \( \Sigma = \{0, 1\} \):
  \[
  L = \{ w \in \Sigma^* \mid w \text{ has the same number of } 0\text{s and } 1\text{s } \}
  \]
- This language is also not regular, but it is context-free.
- How might we design a TM for it?
A Caveat
A Caveat

... 0 0 1 1 1 0 1 1 0 ...
A Caveat
A Caveat
A Caveat

| ... | 0 | 0 | 1 | 1 | 1 | 0 | ... |
A Caveat
A Caveat
A Caveat

... 0 0 1 1 1 0 ...

...
A Caveat
A Caveat
A Caveat

How do we know that this blank isn't one of the infinitely many blanks after our input string?
A Caveat
A Caveat

... 0 1 1 0 ...

...
A Caveat
A Caveat
A Caveat
| ... |   |   |   | 1 | 1 | 0 |   | ... |

A Caveat
A Caveat

How do we know that this blank isn't one of the infinitely many blanks after our input string?
A Caveat

... 1 1 0 ...

A Caveat

How do we know that this blank isn't one of the infinitely many blanks after our input string?
One Solution
One Solution

... \times 0 0 1 1 1 1 1 0 ...
One Solution

... \times 00111110 ...
One Solution
One Solution

\[ \ldots \times 0 0 \times 1 1 1 0 \ldots \]
One Solution

\[ \cdots \times 0 0 \times 1 1 1 0 \cdots \]
One Solution

<table>
<thead>
<tr>
<th>...</th>
<th>×</th>
<th>0</th>
<th>0</th>
<th>×</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>0</th>
<th>...</th>
</tr>
</thead>
</table>
One Solution
One Solution

... × 0 0 × 1 1 1 0 ...

...
One Solution
One Solution
One Solution

… × × 0 × 1 1 1 1 0 …
One Solution

... × × 0 × 1 1 1 1 0 ...
One Solution
One Solution

... × × 0 × × 1 1 0 ...
One Solution

... × × 0 × × 1 1 0 ...

One Solution
One Solution

... × × 0 × × 1 1 0 ...

One Solution
One Solution

... × × 0 × × 1 1 0 ...
One Solution

... × × 0 × × 1 1 0 ...
| ... | × × × × × × 1 1 0 | ... |
One Solution

... × × × × × × 1 1 0 ...
One Solution
One Solution

\[ \ldots \times \times \times \times \times \times \times 1 0 \ldots \]
One Solution
Find 0/1

Start

0 → ×, R

... 0 0 1 1 1 1 1 0 0 ...

...
Find 0/1

0 → x, R

... × 0 1 1 1 1 1 0 0 ...
Find 0/1

0 → ×, R

Find 1

... × 0 1 1 1 1 1 0 0 ...

start
Find 0/1

0 → ×, R

Find 1

0 → 0, R

... x 0 1 1 1 1 1 0 0 ...

...
Find 0/1

0 → ×, R

Find 1

1 → ×, L

0 → 0, R

... × 0 1 1 1 1 1 1 0 0 ...

start
Find 0/1

0 → ×, R

Find 1

1 → ×, L

0 → 0, R

... × 0 × 1 1 1 1 0 0 ...

start
Find 0/1

Start

Find 0

0 → x, R

Find 1

0 → 0, R

1 → x, L

Go home

... × 0 × 1 1 1 1 0 0 …
start

Find 0/1

0 → x, R

Find 1

1 → x, L

0 → 0, R

Go home

0 → 0, L

1 → 1, L

x → x, L

... x 0 x 1 1 1 1 0 0 ...
Find 0/1

0 → ×, R

Find 1

1 → ×, L

0 → 0, R

Go home

0 → 0, L
1 → 1, L
× → ×, L

... × 0 × 1 1 1 1 0 0 ...
Find 0/1

0 → x, R

Find 1

1 → x, L

0 → 0, R

Go home

0 → 0, L
1 → 1, L
x → x, L

... × 0 × 1 1 1 1 0 0 ...
Find 0/1

0 → ×, R
0 → 0, R

Find 1

1 → ×, L

□ → □, R

0 → 0, L
1 → 1, L
x → x, L

Go home

0 → 0, L
1 → 1, L
x → x, L

... × 0 × 1 1 1 1 0 0 ...
Find 0/1

Find 1

Go home

0 → 0, L
1 → 1, L
× → ×, L

0 → ×, R
1 → ×, L
0 → 0, R

... x 0 x 1 1 1 1 0 0 ...
start

- \( x \rightarrow x, R \)
- \( 0 \rightarrow x, R \)
- \( 1 \rightarrow x, L \)
- \( 0 \rightarrow 0, R \)
- \( \square \rightarrow \square, R \)
- \( 0 \rightarrow 0, L \)
- \( 1 \rightarrow 1, L \)
- \( x \rightarrow x, L \)

---

... X 0 X 1 1 1 1 0 0 ...
Find 0/1

Find 1

Go home

\( x \rightarrow x, R \)

\( 0 \rightarrow x, R \)

\( 1 \rightarrow x, L \)

\( 0 \rightarrow 0, R \)

\( 0 \rightarrow 0, L \)

\( 1 \rightarrow 1, L \)

\( x \rightarrow x, L \)

\( \square \rightarrow \square, R \)
\begin{itemize}
\item \textit{x → \textit{x}, R}
\item \textit{\textbf{Find 0/1}}
\item \textit{\textbf{0 → \textit{x}, R}}
\item \textit{\textbf{Find 1}}
\item \textit{\textbf{1 → \textit{x}, L}}
\item \textit{\textbf{0 → 0, R}}
\item \textit{\textbf{Go home}}
\item \textit{\textbf{0 → 0, L}}
\item \textit{\textbf{1 → 1, L}}
\item \textit{\textbf{x → \textit{x}, L}}
\end{itemize}
Find 0/1

\[ \times \rightarrow \times, R \]

Find 1

\[ 0 \rightarrow \times, R \]

\[ 1 \rightarrow \times, L \]

Go home

\[ 0 \rightarrow 0, L \]

\[ 1 \rightarrow 1, L \]

\[ \times \rightarrow \times, L \]

...\[ \times \times \times \times \] 1 1 1 1 0 0 ...

...
Find 0/1

0 \rightarrow \times, R

\[ \times \rightarrow \times, R \]

\[ \Box \rightarrow \Box, R \]

Go home

0 \rightarrow 0, L

1 \rightarrow 1, L

\times \rightarrow \times, L

Find 1

0 \rightarrow 0, R

\times \rightarrow \times, R

1 \rightarrow \times, L

\[ \times \times \times \times 1 1 0 0 \ldots \]
Find 0/1

- Start: \(x \rightarrow x, R\)
- 1: \(1 \rightarrow x, R\)
- 0: \(0 \rightarrow x, R\)
- \(\square \rightarrow \square, R\)
- Find 1

- 0: \(0 \rightarrow 0, R\)
- \(x \rightarrow x, R\)
- 1: \(1 \rightarrow x, L\)

Go home

- 0: \(0 \rightarrow 0, L\)
- 1: \(1 \rightarrow 1, L\)
- \(x \rightarrow x, L\)
start

Find 0/1

Find 0

Find 1

Go home

0 → ×, R
1 → ×, R
× → ×, R
0 → ×, L
1 → ×, L
× → ×, L
0 → 0, L
1 → 1, L
× → ×, L

0 → 0, R
1 → 1, R
× → ×, R
0 → ×, R
1 → ×, R
× → ×, R

... × × × × × 1 0 0 ...
Go home

Find 0

1 → 1, R

\( \times \rightarrow \times, R \)

Find 1

1 → 1, R

0 → 1, R

Find 0/1

\( \times \rightarrow \times, R \)

0 → 1, L

\( \times \rightarrow \times, L \)

Go home

0 → 0, L

1 → 1, L

\( \times \rightarrow \times, L \)

\[ \ldots \]

\( \times \times \times \times \times \times \times \times \times 1 \times \times 0 \)

\[ \ldots \]
Go home

Find 0

1 → 1, R
\( \times \rightarrow \times, R \)

Find 1

1 → \( \times \), L

\( 0 \rightarrow \times, L \)

Find 0/1

\( \times \rightarrow \times, R \)

Go home

0 → 0, L
1 → 1, L
\( \times \rightarrow \times, L \)

start

\( x \rightarrow x, R \)

…

\[ \times \times \times \times \times \times \times 1 \times \times 0 \]

…
Start

Find 0/1

Find 0

Find 1

Go home

0 → 0, L
1 → 1, L
x → x, L

0 → x, R
1 → x, R
x → x, R

□ → □, R

0 → 0, R

0 → x, R
1 → x, R

... x x x x x x x 1 x x 0 ...
start

x → x, R

Find 0/1

x → x, R

Find 1

Find 0

1 → x, R

0 → x, L

Go home

0 → 0, L

1 → 1, L

x → x, L

□ → □, R

1 → 1, R

0 → 0, R

x → x, R

0 → 0, R

1 → 1, R

× → ×, R

× → ×, L

0 → 0, R

1 → 1, R

× → ×, L

× → ×, R

…

× × × × × × × × × × × × ...

…
...  

X X X X X X X X X X X ...

---

Find 0/1

Find 0

Find 1

Go home

Start

x → x, R

1 → x, R

x → x, R

0 → x, L

□ → □, R

0 → 0, L

1 → 1, L

x → x, L

1 → 1, R

x → x, R

0 → x, R

0 → x, L

1 → x, L

0 → 0, R

x → x, R

1 → x, L

0 → x, R

...
<table>
<thead>
<tr>
<th>Action</th>
<th>Movement</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>×, L</td>
</tr>
<tr>
<td>1</td>
<td>×, R</td>
</tr>
<tr>
<td>×</td>
<td>×, R</td>
</tr>
<tr>
<td>Find 0</td>
<td>1 → 1, R</td>
</tr>
<tr>
<td></td>
<td>× → ×, R</td>
</tr>
<tr>
<td>Find 1</td>
<td>0 → ×, L</td>
</tr>
<tr>
<td></td>
<td>1 → ×, L</td>
</tr>
<tr>
<td>Find 0/1</td>
<td>x → x, R</td>
</tr>
<tr>
<td>Go home</td>
<td>0 → 0, L</td>
</tr>
<tr>
<td></td>
<td>1 → 1, L</td>
</tr>
<tr>
<td></td>
<td>× → ×, L</td>
</tr>
</tbody>
</table>

Diagram:
- Start at Find 0/1.
- Move to Find 0 if you see 0, else move to Find 1.
- From Find 0, move right if you see 1, loop back if you see ×.
- From Find 1, move left if you see 1, loop back if you see ×.
- Move to Go home if you see 0, else loop back to Find 0/1.

Array:
```
... x x x x x x x x x x ...
```
Remember that all missing transitions implicitly reject.
Constant Storage

- Sometimes, a TM needs to remember some additional information that can't be put on the tape.
- In this case, you can use similar techniques from DFAs and introduce extra states into the TM's finite-state control.
- The finite-state control can only remember one of finitely many things, but that might be all that you need!