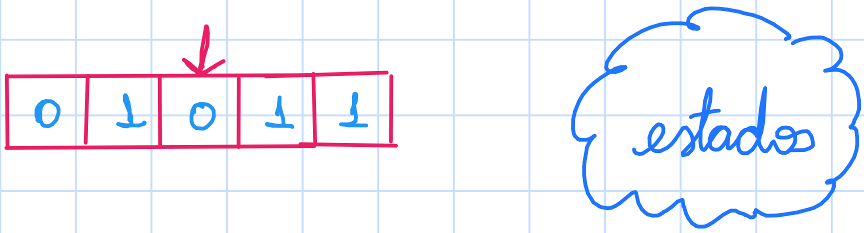
# Máquinas de Turing

## Máquinas de Turing

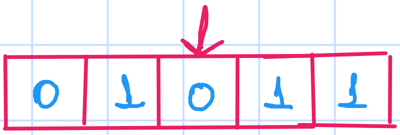
### Autômato Finito (não) Determinístico

cadeia de entrada

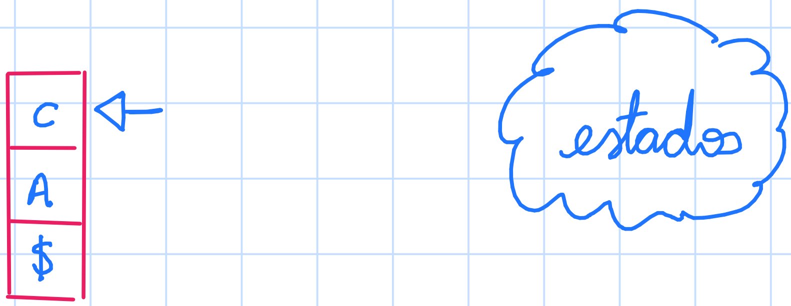



### Autômato com Pilha

- cadeia de entrada




- uma pilha



# Máquina de Turing

- uma "fita"

infinita - à direita

estados

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 0 | 1 | 1 | $\sqcup$ | $\sqcup$ | $\sqcup$ | ... |

→ Alfabeto da Fita : $\Gamma$

→ $\Gamma \supset \Sigma$

símbolo de branco

→ $\sqcup \notin \Sigma$ e $\sqcup \in \Gamma$

## Configuração Inicial

Fita

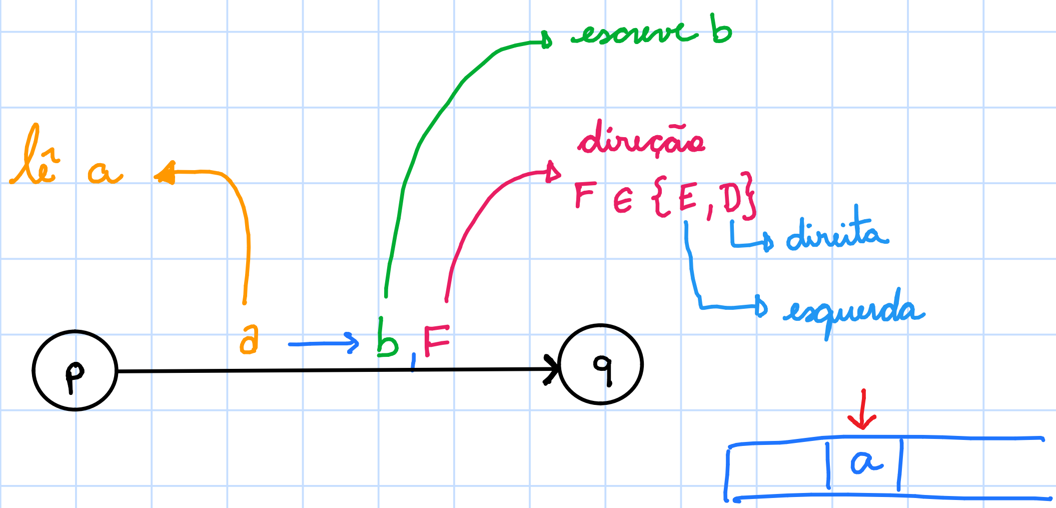
$$\underbrace{0 \mid 1 \mid 0 \mid 1 \mid 1}_{\text{cadeia de entrada}} \mid \underbrace{\sqcup \mid \sqcup \mid \sqcup \mid \dots}_{\text{células em branco}}$$

estado inicial → s

## Estados Finais

* Dois estados finais: aceita e rejeita.

# Função de transição

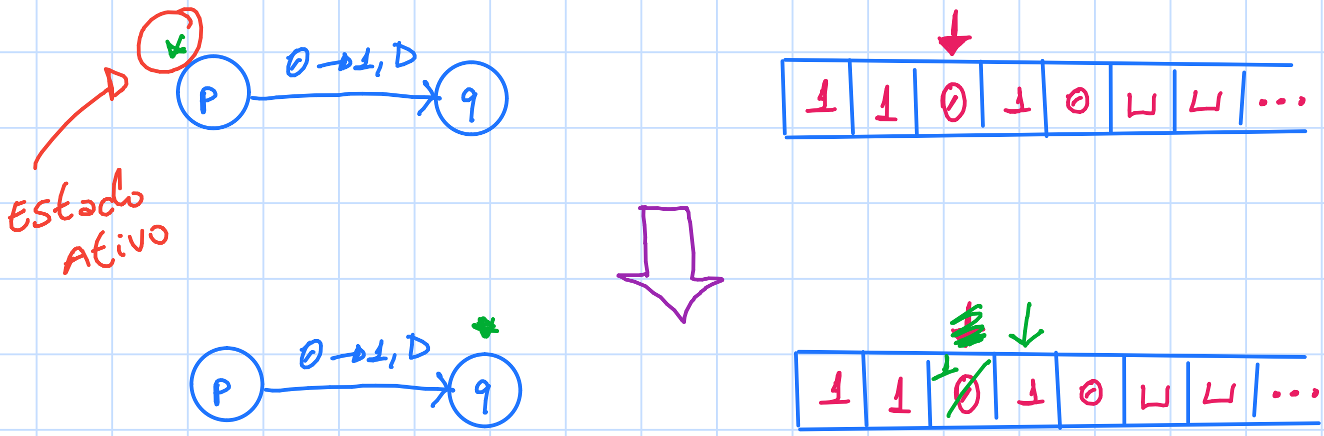


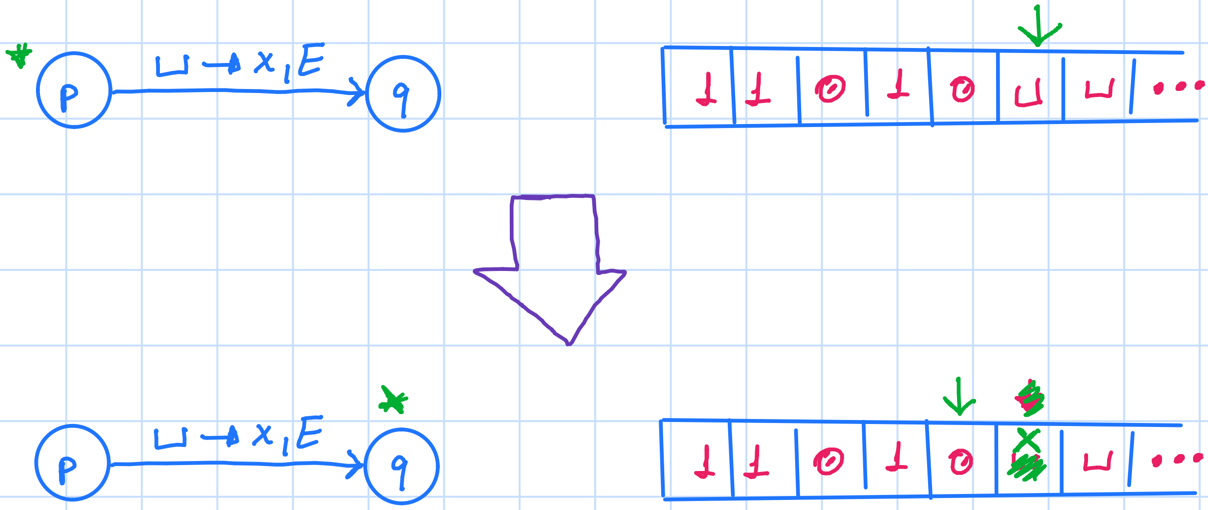

Em cada passo:

① lê o símbolo atual
② atualiza a célula (i.e., escreve)
③ move para a direita ou esquerda

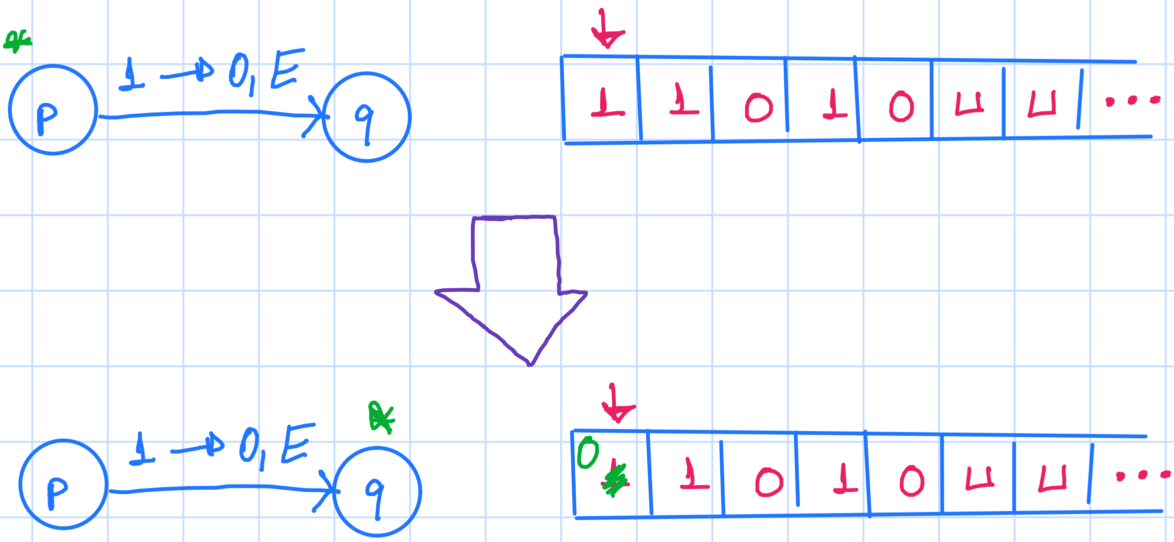
* Determinística!

## Exemplos

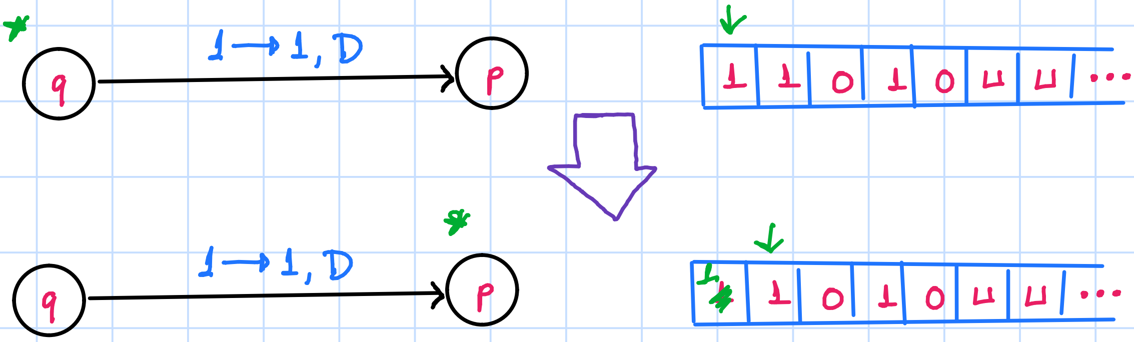




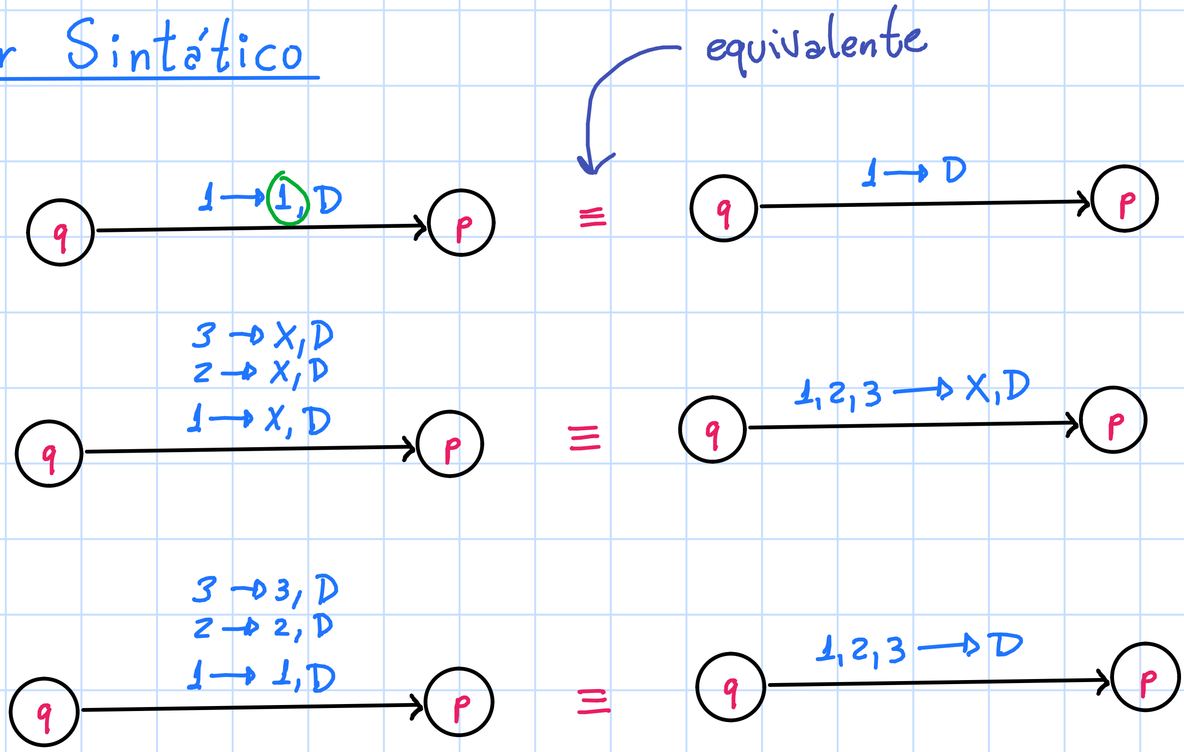

Se você está na borda esquerda e tentou mover-se à esquerda, a cabeça "patina" e n̄ sai do lugar.



não quer atualizar a célula?




## Açúcar Sintático




## Computação

* Aceita
  * Para o processamento e aceita.
* Rejeita
  * Para o processamento e rejeita.
* Loop
  * Máquina não para

# Máquina de Turing
# (Formal)

Uma máquina de Turing é uma 7-upla

$$(Q, \Sigma, \Gamma, \delta, q_0, q_{aceita}, q_{rejeita}),$$

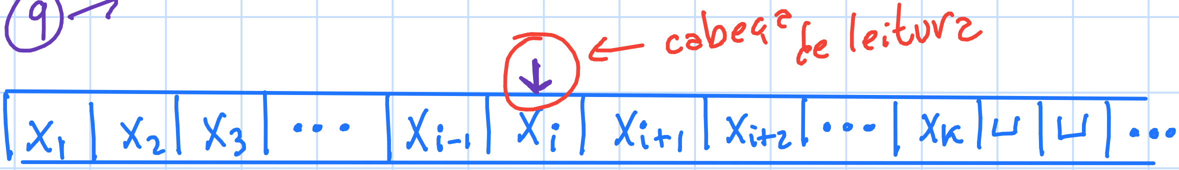
Onde:

* $Q$ é um conjunto finito de elementos chamados estados.
* $\Sigma$ é o alfabeto de entrada $(\sqcup \notin \Sigma)$
* $\Gamma$ é o alfabeto da fita, onde $\Sigma \subset \Gamma$ e $\sqcup \in \Gamma$
* $\delta : Q \times \Gamma \longrightarrow Q \times \Gamma \times \{E, D\}$ é a função de transição
* $q_0 \in Q$ é o estado inicial
* $q_{aceita} \in Q$ é o estado de aceitação
* $q_{rejeita} \in Q \setminus \{q_{aceita}\}$ é o estado de rejeição

## Configuração Instantânea (C.I.)

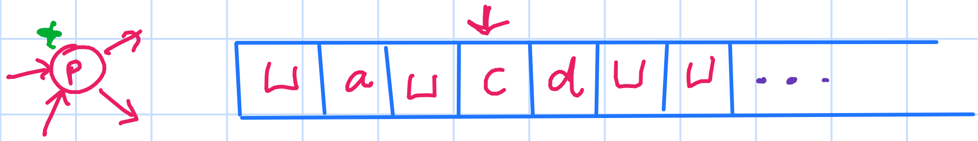
$Q \cap \Gamma = \emptyset$


$X_1 X_2 X_3 \cdots X_{i-1}\, q\, X_i X_{i+1} X_{i+2} \cdots X_k$




- $q$ estado atual.
- $X_i$ é a célula na qual a cabeça de leitura se encontra.

**Exemplo**



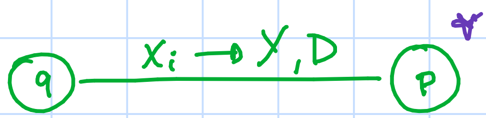

C.I. $\quad \sqcup a \sqcup p\, c\, d$

* importante : $\Gamma \cap Q = \emptyset$

## Fluxo de Execução de uma TM

### * Movendo p/ a direita

$\delta(q, X_i) = (p, Y, D)$




$$X_1 X_2 X_3 \cdots X_{i-1}\, q\, X_i X_{i+1} \cdots X_k \vdash X_1 X_2 X_3 \cdots X_{i-1}\, Y\, p\, X_{i+1} \cdots X_k$$

Caso especial: $i = k$

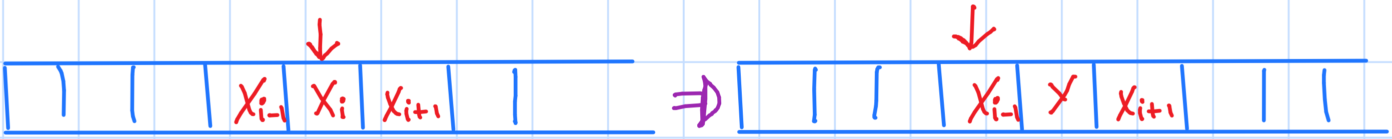
$$X_1X_2X_3\cdots X_{k-1}\,q\,X_k \vdash X_1X_2X_3\cdots X_{k-1}\,Y\,p\,\sqcup$$


## * Movendo p/ a esquerda

$$\delta(q, X_i) = (p, Y, E)$$

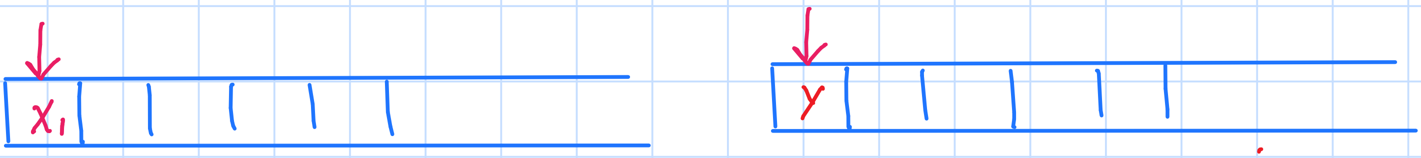
$$q \xrightarrow{X_k \to Y,\,E} p$$

$$X_1X_2X_3\cdots X_{i-1}\,q\,X_i\,X_{i+1}\cdots X_k \vdash X_1X_2X_3\cdots p\,X_{i-1}\,Y\,X_{i+1}\cdots X_k$$




Caso especial: $i = 1$

$$q\,X_1X_2X_3\cdots X_k \vdash p\,Y\,X_2X_3\cdots X_k$$




Dadas configurações Instantâneas A e B, denotamos por $A \vdash^* B$, se existe uma sequência de zero ou mais configurações instantâneas tais que

$$A = I_1 \vdash I_2 \vdash I_3 \vdash \cdots \vdash I_n = B$$

Nota: $A \vdash^* A$ para qualquer configuração instantânea.

## Máquina de Turing:

- parar e aceitar
- parar e rejeitar
- entrar em loop

Uma MT $(Q, \Sigma, \Gamma, \delta, q_0, q_a, q_r)$ aceita uma cadeia $w \in \Sigma^*$ se

$$q_0 w \vdash^* q_a$$

Uma MT $M = (Q, \Sigma, \Gamma, \delta, q_0, q_a, q_r)$ reconhece uma linguagem L se

$$L = \{ w \in \Sigma^* : q_0 w \vdash^* q_a \}$$

* Linguagem de M
* Linguagem reconhecida por M
* $L(M) = L$

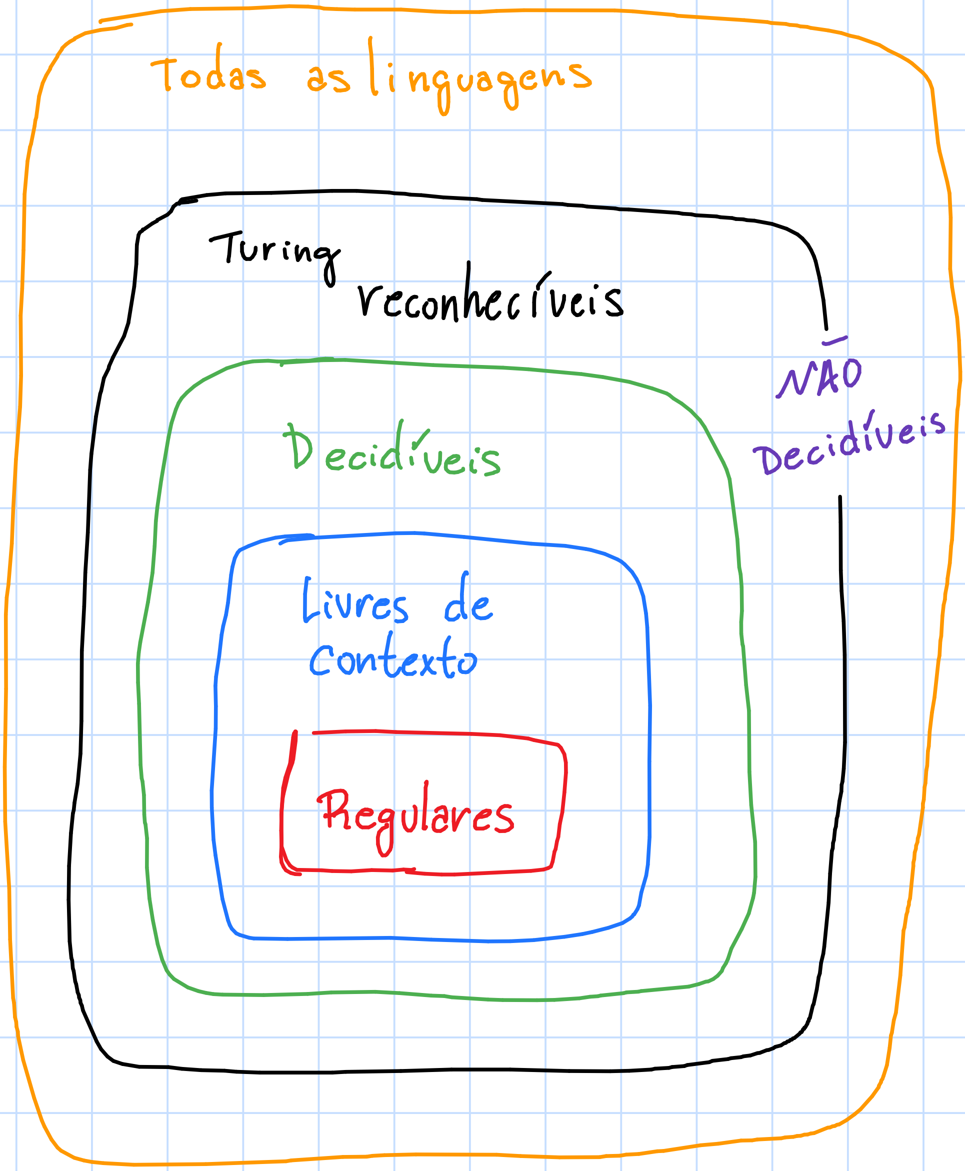
se $w \notin L$ é possível que a máquina entre em

**LOOP**

Uma MT $M = (Q, \Sigma, \Gamma, \delta, q_0, q_a, q_r)$ decide uma linguagem L se M para para qualquer $w \in \Sigma^*$.

Uma linguagem é (turing) reconhecível (tbm chamada de recursivamente enumerável) se alguma máquina de turing a reconhece.

Uma linguagem é (turing) decidível (tbm chamada de recursiva) se alguma máquina de turing a decide.



# Exemplo

$$L = \{a^n b^n c^n : n \geq 0\}$$

| ↓ | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| a | a | a | b | b | b | c | c | c | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ... |

| | ↓ | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| X | a | a | b | b | b | c | c | c | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ... |

| | | | ↓ | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| X | a | a | b | b | b | c | c | c | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ... |

| | | | | ↓ | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| X | a | a | Y | b | b | c | c | c | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ... |

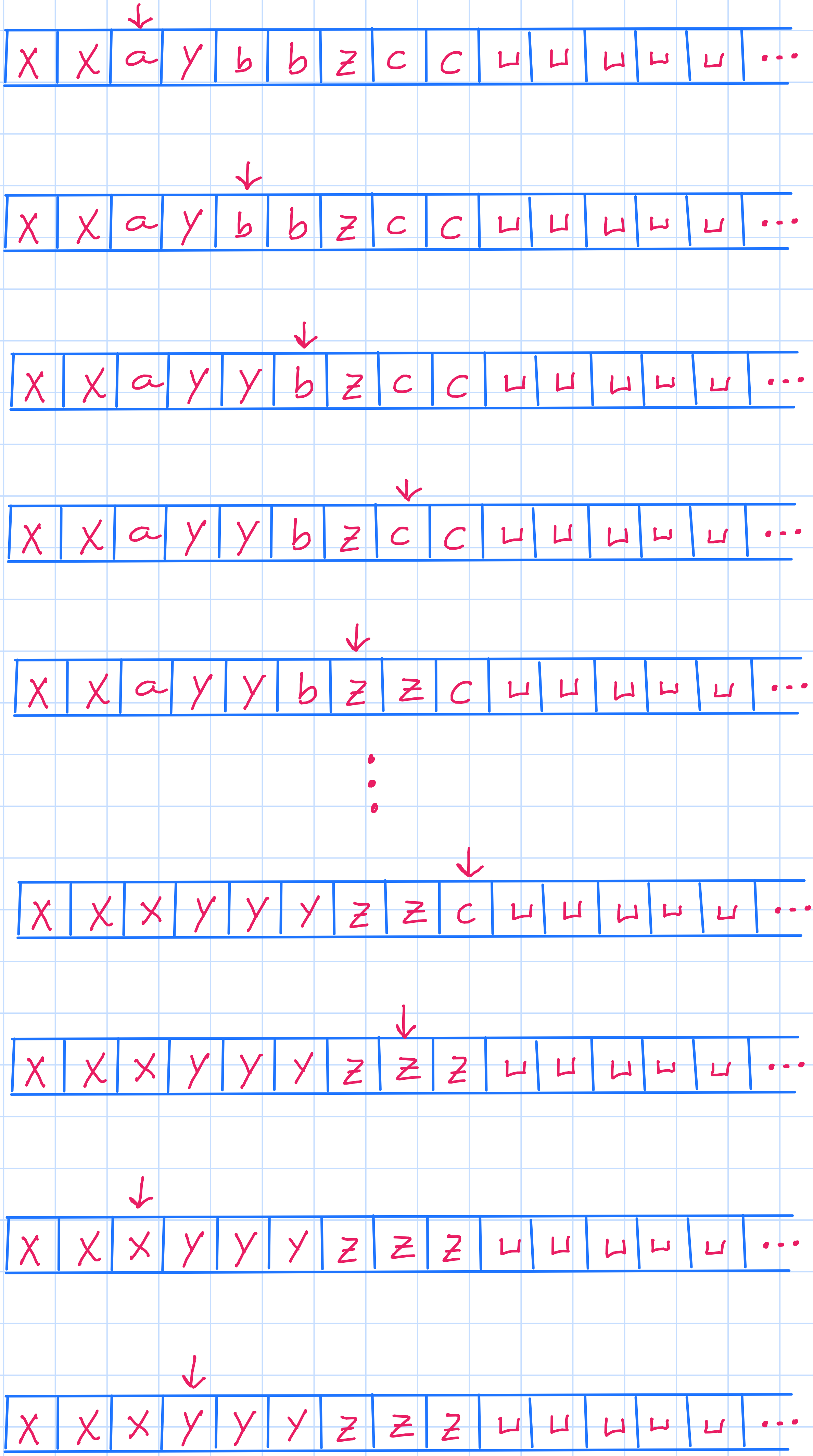
| | | | | | | ↓ | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| X | a | a | Y | b | b | c | c | c | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ... |

| | | | | | ↓ | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| X | a | a | Y | b | b | Z | c | c | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ... |

| ↓ | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| X | a | a | Y | b | b | Z | c | c | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ... |

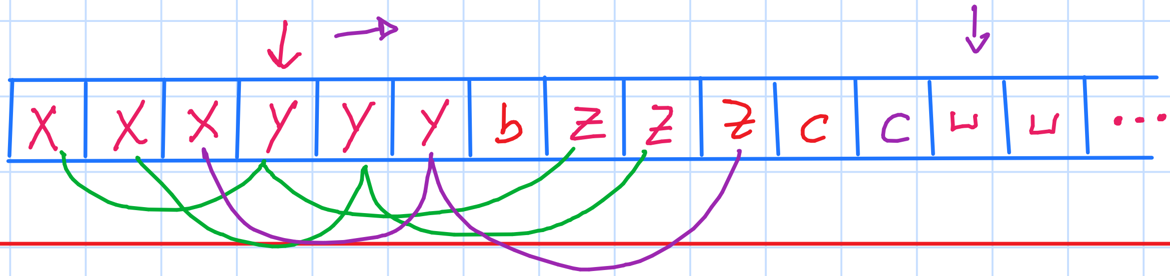
| | ↓ | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| X | a | a | Y | b | b | Z | c | c | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ... |

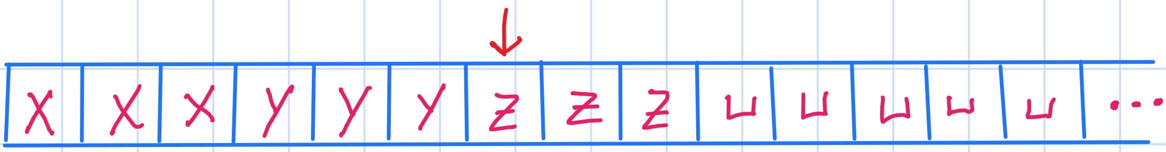
| | ↓ | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| X | a | a | Y | b | b | Z | c | c | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ... |

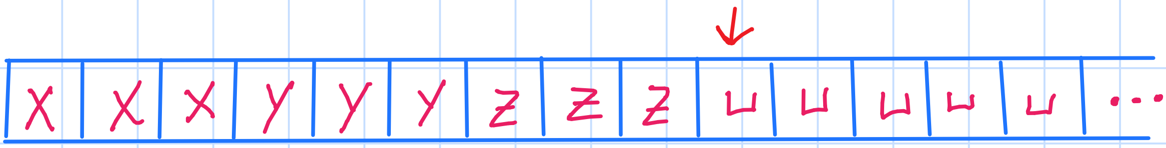

x x a y b b z c c ⊔ ⊔ ⊔ ⊔ ⊔ …
x x a y b b z c c ⊔ ⊔ ⊔ ⊔ ⊔ …
x x a y y b z c c ⊔ ⊔ ⊔ ⊔ ⊔ …
x x a y y b z c c ⊔ ⊔ ⊔ ⊔ ⊔ …
x x a y y b z z c ⊔ ⊔ ⊔ ⊔ ⊔ …
x x x y y y z z c ⊔ ⊔ ⊔ ⊔ ⊔ …
x x x y y y z z z ⊔ ⊔ ⊔ ⊔ ⊔ …
x x x y y y z z z ⊔ ⊔ ⊔ ⊔ ⊔ …
x x x y y y z z z ⊔ ⊔ ⊔ ⊔ ⊔ …

Aceita?

ainda não ...

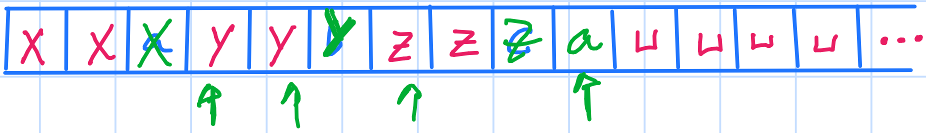


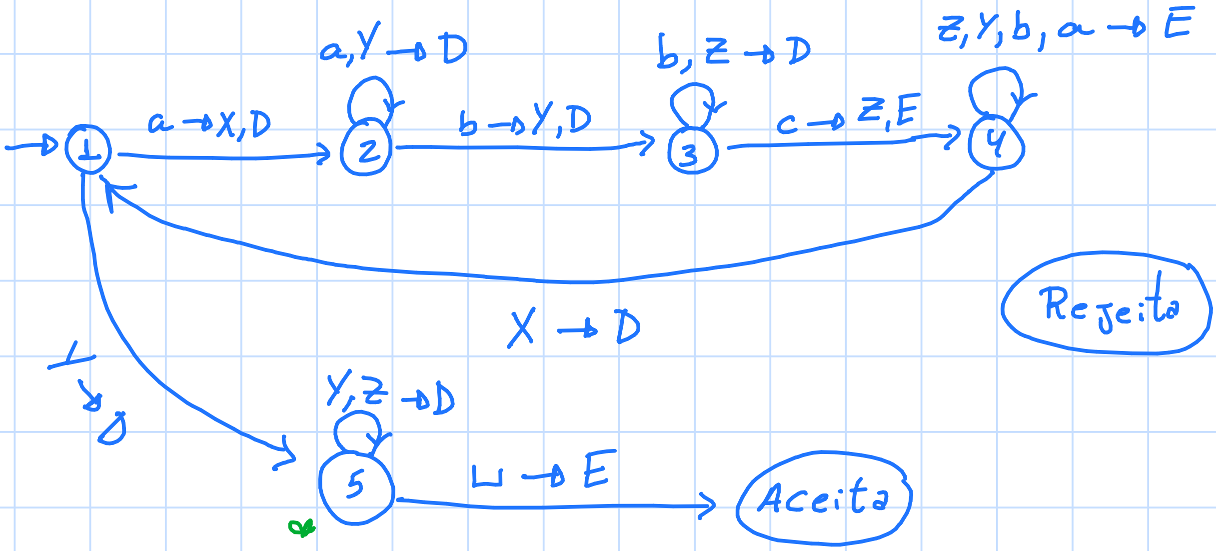




Aceita!

## Diagrama de Estados

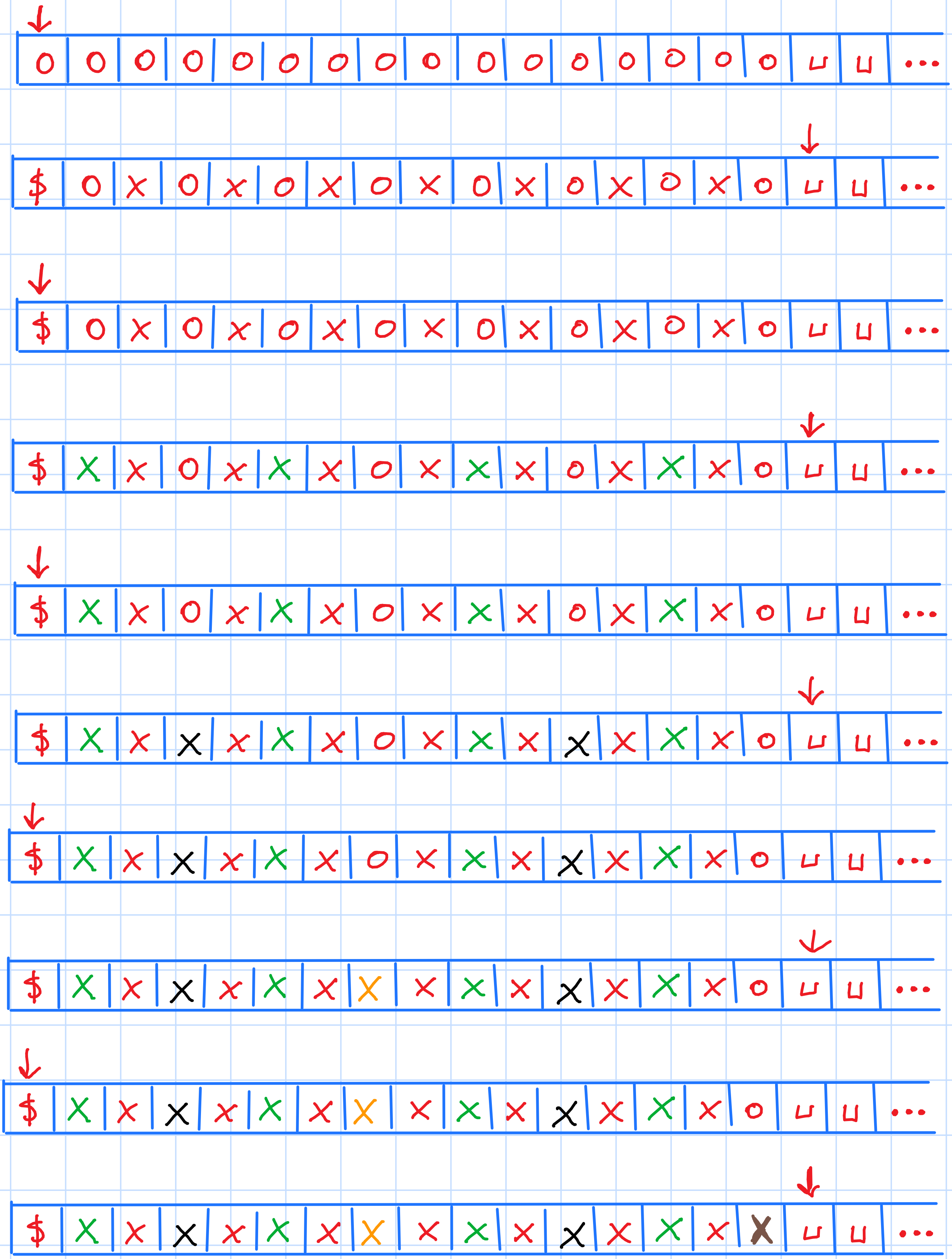



## Convenção

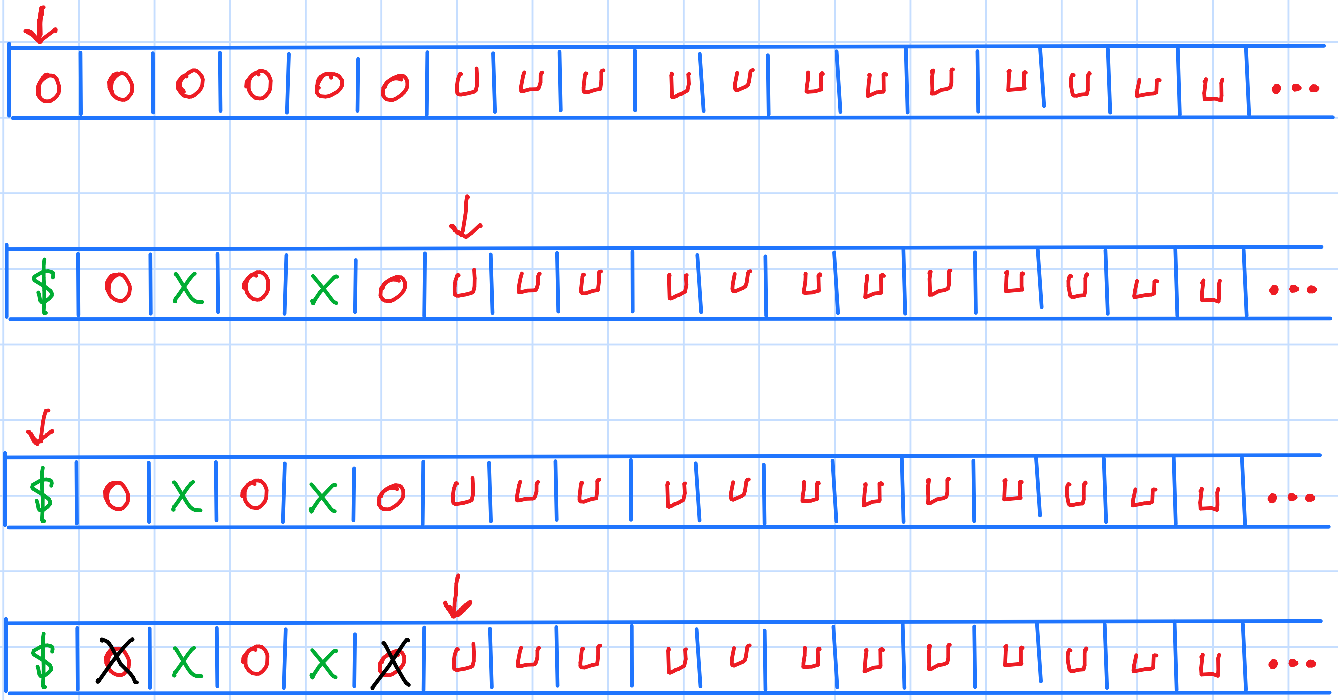
transições omitidas levam para o estado de rejeição

$$A = \{0^{2^n} \mid n \geq 0\}$$

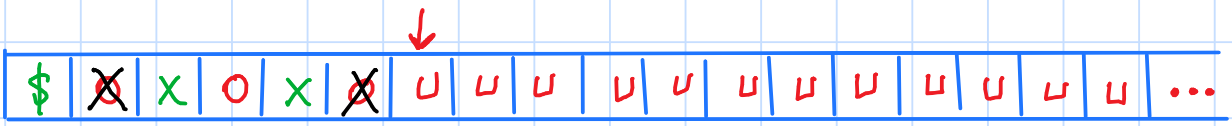


Aceita!

$\omega = 0^6$

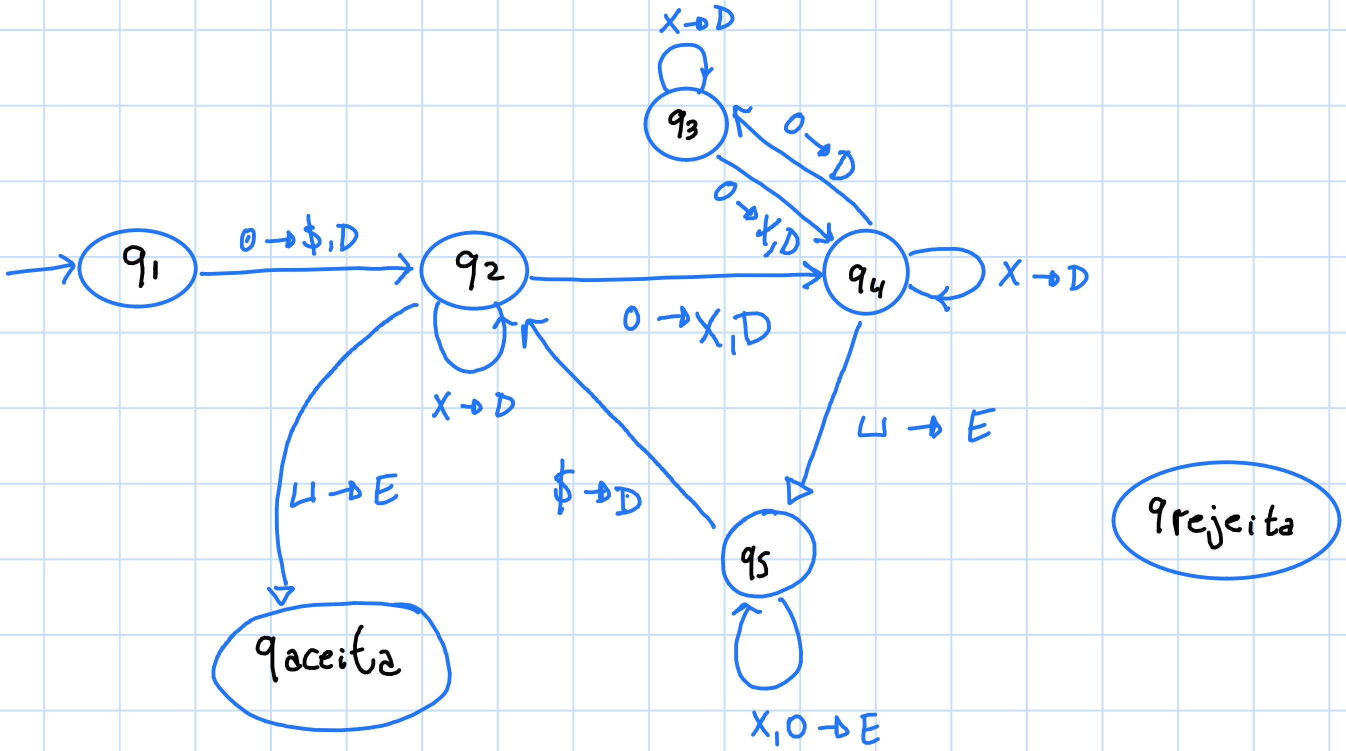


Rejeita!



## Estados

- $q_1$ estado inicial
- $q_2$ vimos exatamente 1 valor zero e ele ñ foi marcado
- $q_4$ vimos um número par de zeros e o último zero foi marcado
- $q_3$ vimos um número ímpar de zeros, maior que 1, e o último zero ñ foi marcado
- $q_5$ retrocede a cabeça de leitura p/ o início.




| | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $ | X | X | 0 | X | X | X | 0 | X | 0 | X | 0 | X | 0 | X | 0 | ⊔ | ⊔ | ... |

| | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ⊔ | ⊔ | ... |

| | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $ | X | 0 | X | 0 | X | 0 | X | 0 | X | 0 | X | 0 | X | 0 | X | ⊔ | ⊔ | ... |

Máquina de Turing tem o mesmo poder computacional de um computador moderno.

→ Programar uma MT por diagrama é difícil.

# Computador

código de máquina

0100100

Código Assembly

ADD $R_1, R_2, R_3$

Código em C

$i = (z+k) * i$

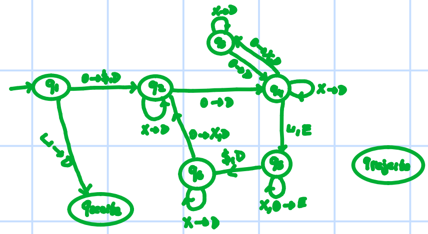
Algoritmos

if $S \cap T \neq \emptyset$

(sem detalhe de implementação)

# MT

diagrama de transições



Descrição nível intermediário

Fala sobre como funciona a cabeça de leitura e a manipulação da fita

Descrição Alto nível

Descreve a MT sem falar sobre a estrutura da MT.

## Descrição em nível intermediário (da MT anterior)

1. percorra a fita marcando os zeros de forma intercalada.
2. Se no passo 1 encontramos apenas um 0, aceite a cadeia.
3. Se no passo 1 encontramos mais de um 0 e o # de 0's encontrados foi ímpar, rejeita a cadeia.
4. Retorne a cabeça para o extremo esquerdo da fita
5. Volte para o passo 1 e repita o processo.

$B = \{ w\#w \mid w \in \{0,1\}^* \}$

| ↓ | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 1 | 1 | # | 0 | 1 | 0 | 0 | 1 | 1 | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ... |

| ↓ | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 1 | 1 | # | 0 | 1 | 0 | 0 | 1 | 1 | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ... |

| | ↓ | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $\dot{0}$ | 1 | 0 | 0 | 1 | 1 | # | 0 | 1 | 0 | 0 | 1 | 1 | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ... |

$\Gamma = \{ 0, \dot{0}, \dots \}$

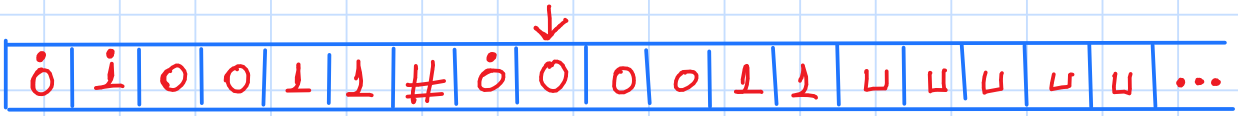
| | | | | | | | ↓ | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $\dot{0}$ | 1 | 0 | 0 | 1 | 1 | # | 0 | 1 | 0 | 0 | 1 | 1 | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ... |

| | | | | | | ↓ | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $\dot{0}$ | 1 | 0 | 0 | 1 | 1 | # | $\dot{0}$ | 1 | 0 | 0 | 1 | 1 | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ... |

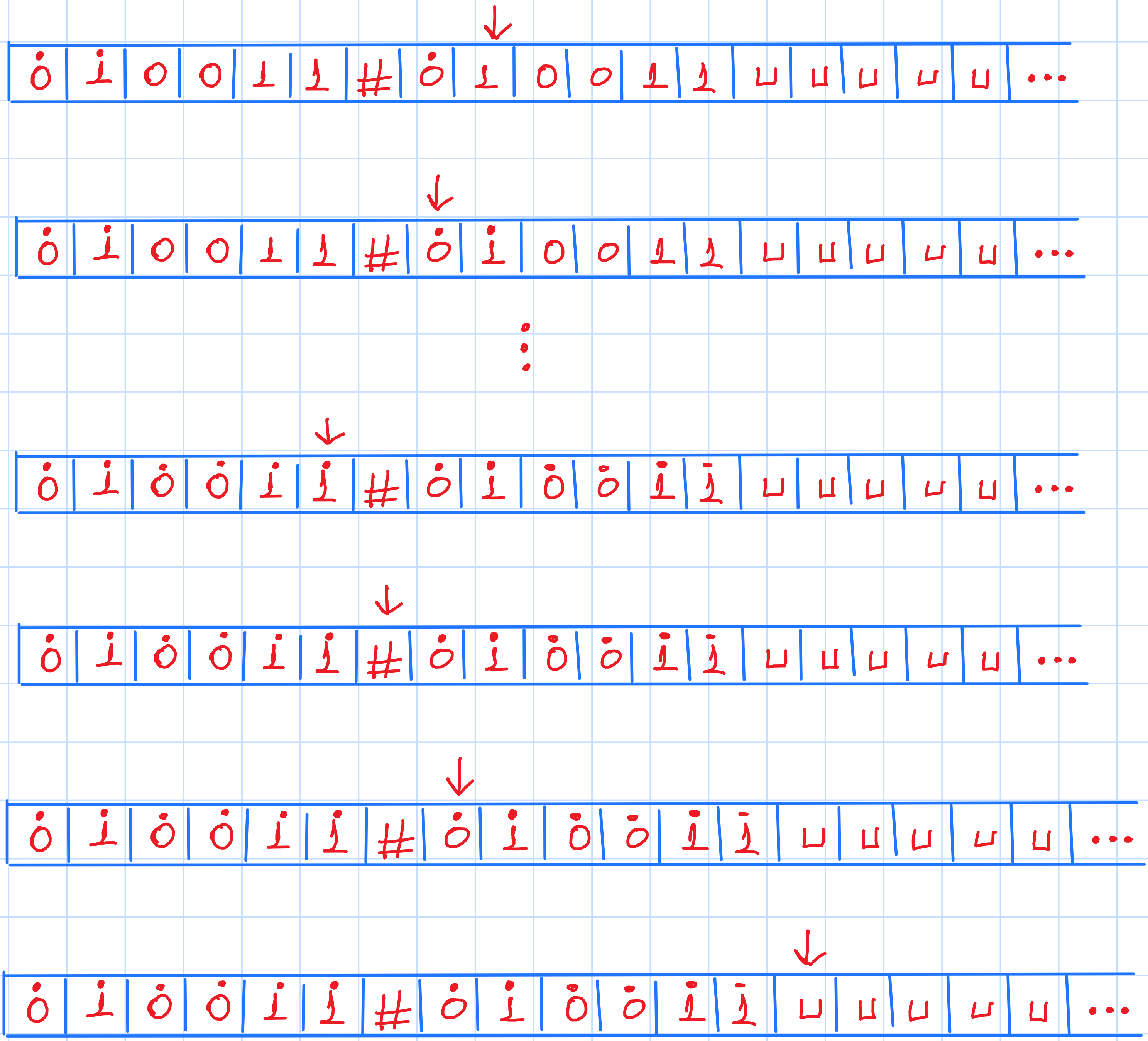
| ↓ | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $\dot{0}$ | 1 | 0 | 0 | 1 | 1 | # | $\dot{0}$ | 1 | 0 | 0 | 1 | 1 | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ... |

| | ↓ | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $\dot{0}$ | 1 | 0 | 0 | 1 | 1 | # | $\dot{0}$ | 1 | 0 | 0 | 1 | 1 | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ... |

| | | ↓ | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $\dot{0}$ | $\dot{1}$ | 0 | 0 | 1 | 1 | # | $\dot{0}$ | 1 | 0 | 0 | 1 | 1 | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ... |



Rejeita



Aceita!

# Estados

$q_1$ : não marquei nada

$q_2$ : marquei um 0 e estou na primeira metade da cadeia

$q_3$ : marquei um 1 e estou na primeira metade da cadeia

| | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $\dot{0}$ | $\dot{1}$ | $\dot{0}$ | 0 | 1 | 1 | # | $\dot{0}$ | $\dot{1}$ | $\dot{0}$ | 0 | 1 | 1 | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ... |

$q_4$ : marquei um 0 e estou na segunda metade da cadeia

$q_5$ : marquei um 1 e estou na segunda metade da cadeia

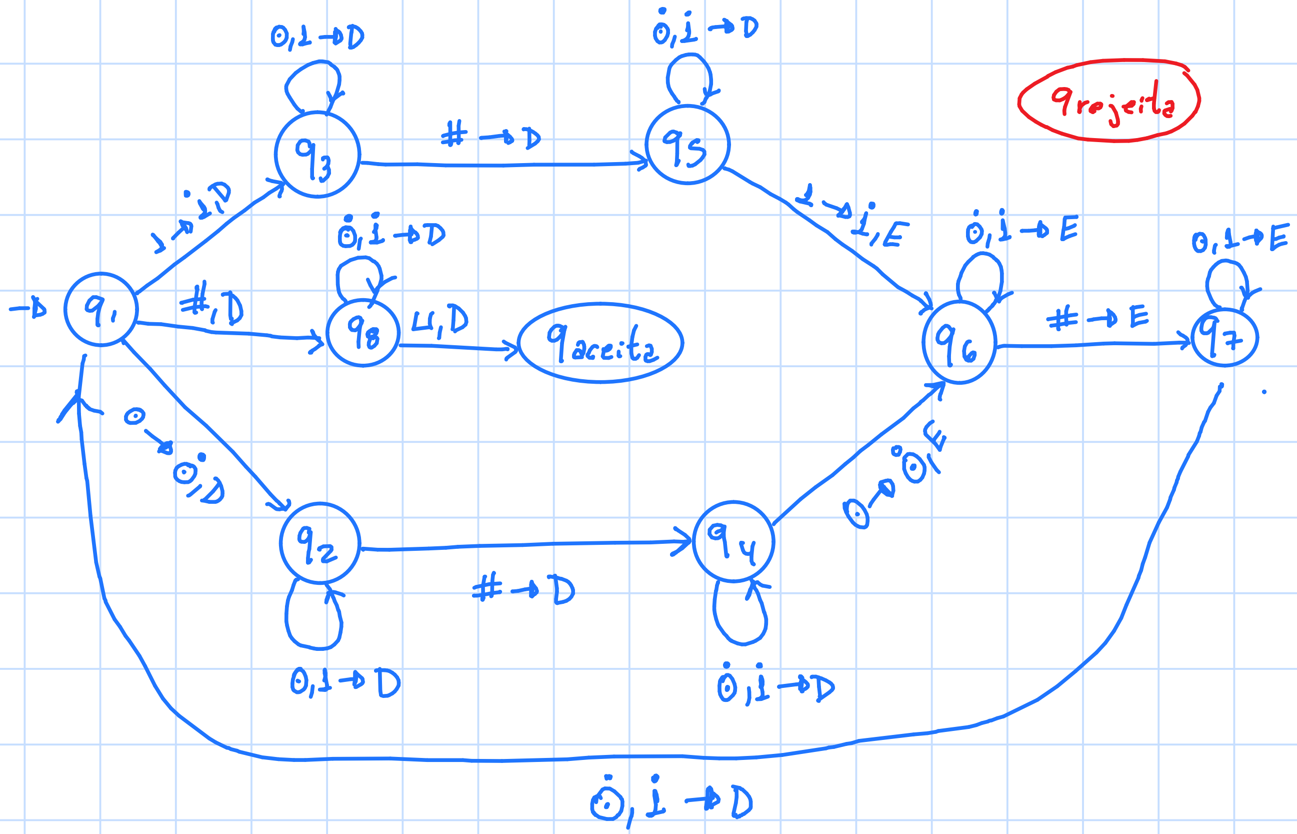
$q_6$ : rebobinando e na segunda metade

$q_7$ : rebobinando e na primeira metade

$q_8$ : checando se não sobrou símbolos na segunda metade.

$q_{aceita}$

$q_{rejeita}$




| | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $\dot{0}$ | $\dot{1}$ | $\dot{0}$ | 0 | 1 | 1 | # | $\dot{0}$ | $\dot{1}$ | $\dot{0}$ | 0 | 1 | 1 | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ... |

$q_1$ : não marquei nada
$q_2$ : marquei um 0 e estou na primeira metade da cadeia
$q_3$ : marquei um 1 e estou na primeira metade da cadeia
$q_4$ : marquei um 0 e estou na segunda metade da cadeia
$q_5$ : marquei um 1 e estou na segunda metade da cadeia
$q_6$ : rebobinando e na segunda metade
$q_7$ : rebobinando e na primeira metade
$q_8$ : checando se não sobrou símbolos na segunda metade.

# Descrição Intermediária

1. Faça zigue-zague ao longo da fita checando posições correspondentes de ambos os lados do símbolo # para verificar se elas contêm o mesmo símbolo. Se elas não contém ou o símbolo # não for encontrado, <u>rejeite</u>. Marque os símbolos cuja correspondência você já verificou.

2. Quando todos os símbolos à esquerda do # tiverem sido marcados, verifique a existência de algum símbolo remanescente à direita do #. Se resta algum, <u>rejeite</u>; caso contrário, <u>aceite</u>.

## Exercício

Descreva uma MT que decida a linguagem

$$C = \{ a^i b^j c^k : i \times j = k \text{ and } i, j, k \geq 1 \}$$

$a^2 b^4 c^8$

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| a | a | b | b | b | b | c | c | c | c | c | c | c | c | $\sqcup$ | $\sqcup$ | ... |

$$i \times j = \underbrace{j + j + j + \cdots + j}_{i}$$

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| a | a | b | b | b | b | c | c | c | c | c | c | c | c | $\sqcup$ | $\sqcup$ | ... |

# Descrição da MT

1. Percorra a cadeia da esquerda para à direita de forma a determinar se ela pertence a linguagem $a^+ b^+ c^+$

2. Retorne a cabeça de leitura para a posição inicial

3. Marque um a e mova a cabeça até encontrar um b. Alternadamente marque um b e um c até que todos os b's tenham sido marcados. Se todos os c's forem marcados e ainda houver b's não marcados, rejeite a cadeia.

4. Desmarque os b's e repita o passo 3 até que todos os a's sejam marcados. Neste caso, verifique se todos os c's estão marcados. Se sim, aceite a cadeia; caso contrário, rejeite.

# Exercício

Descreva uma MT que decida a linguagem

$$E = \{\#x_1\#x_2\#x_3\#\cdots\#x_\ell \, : \, x_i \in \{0,1\}^* \text{ e } x_i \neq x_j\}$$

$w = \#011\#0110\#01\#01$ $w \notin E$

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| # | 0 | 1 | 1 | # | 0 | 1 | 1 | 0 | # | 0 | 1 | # | 0 | 1 | ⊔ | ... |
| # | 0 | 1 | 1 | # | 0 | 1 | 1 | 0 | # | 0 | 1 | # | 0 | 1 | ⊔ | ... |
| #• | 0 | 1 | 1 | # | 0 | 1 | 1 | 0 | # | 0 | 1 | # | 0 | 1 | ⊔ | ... |
| #• | 0 | 1 | 1 | # | 0 | 1 | 1 | 0 | # | 0 | 1 | # | 0 | 1 | ⊔ | ... |
| #• | 0 | 1 | 1 | #• | 0 | 1 | 1 | 0 | # | 0 | 1 | # | 0 | 1 | ⊔ | ... |
| #• | 0+ | 1 | 1 | #• | 0 | 1 | 1 | 0 | # | 0 | 1 | # | 0 | 1 | ⊔ | ... |
| #• | 0+ | 1 | 1 | #• | 0 | 1 | 1 | 0 | # | 0 | 1 | # | 0 | 1 | ⊔ | ... |

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| # | 0 | 1 | 1 | # | 0 | 1 | 1 | 0 | # | 0 | 1 | # | 0 | 1 | ⊔ | … |

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| # | 0 | 1 | 1 | # | 0 | 1 | 1 | 0 | # | 0 | 1 | # | 0 | 1 | ⊔ | … |

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| # | 0 | 1 | 1 | # | 0 | 1 | 1 | 0 | # | 0 | 1 | # | 0 | 1 | ⊔ | … |

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| # | 0 | 1 | 1 | # | 0 | 1 | 1 | 0 | # | 0 | 1 | # | 0 | 1 | ⊔ | … |

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| # | 0 | 1 | 1 | # | 0 | 1 | 1 | 0 | # | 0 | 1 | # | 0 | 1 | ⊔ | … |

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| # | 0 | 1 | 1 | # | 0 | 1 | 1 | 0 | # | 0 | 1 | # | 0 | 1 | ⊔ | … |

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| # | 0 | 1 | 1 | # | 0 | 1 | 1 | 0 | # | 0 | 1 | # | 0 | 1 | ⊔ | … |

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| # | 0 | 1 | 1 | # | 0 | 1 | 1 | 0 | # | 0 | 1 | # | 0 | 1 | ⊔ | … |

SS

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| # | 0 | 1 | 1 | # | 0 | 1 | 1 | 0 | # | 0 | 1 | # | 0 | 1 | ⊔ | … |

# Descrição da máquina

1. Se o primeiro símbolo da fita for #, coloque uma marca nele. Caso contrário, rejeite a cadeia.

2. Mova a cabeça de leitura para à direita até encontrar # e coloque uma marca nele. Se encontrar o símbolo $\sqcup$, apenas $x_1$ estava presente, aceite a cadeia.

3. Fazendo zigue-zague, compare as duas cadeias à direita dos símbolos #'s marcados. Se forem iguais rejeite a cadeia.

4. Mova a marca do # que está mais à direita para o próximo # à direita. Se tal # não existir, a marca deve ser apagada e a marca do # restante deve ser movida para o próximo # à direita deste. Desta vez, se tal # não existir, significa que todas as cadeias foram comparadas, então aceitamos a cadeia.

5. Vá para o passo 3.

# MÁQUINA DE TURING como subrotina

a) Máquina de turing para inserir uma marca no início da fita.

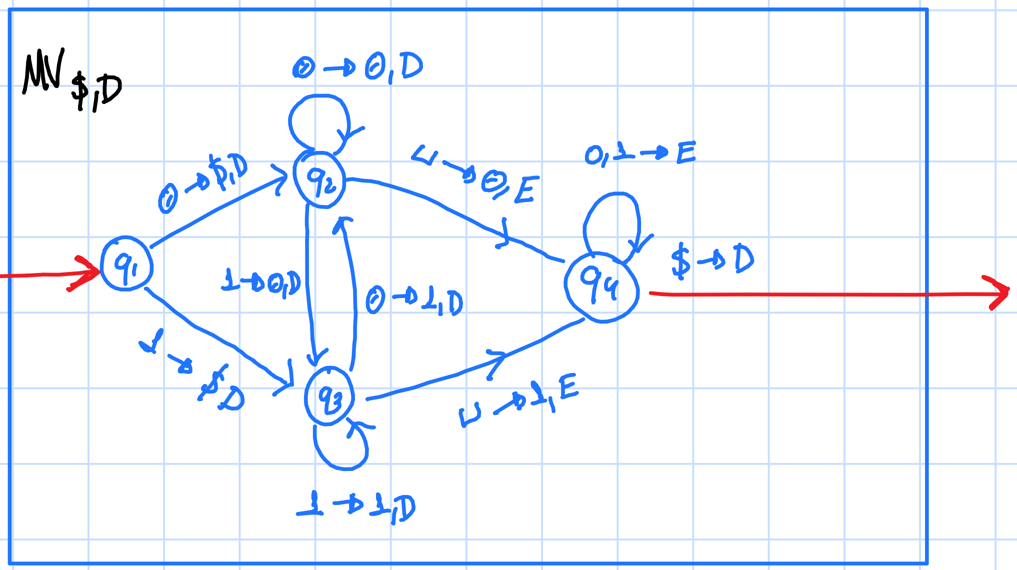
| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ... |

$q_1$ : estado inicial

$q_2$ : último símbolo sobrescrito foi 1

$q_3$ : último símbolo sobrescrito foi 0

$q_4$ : nenhum símbolo foi sobrescrito




Obs:

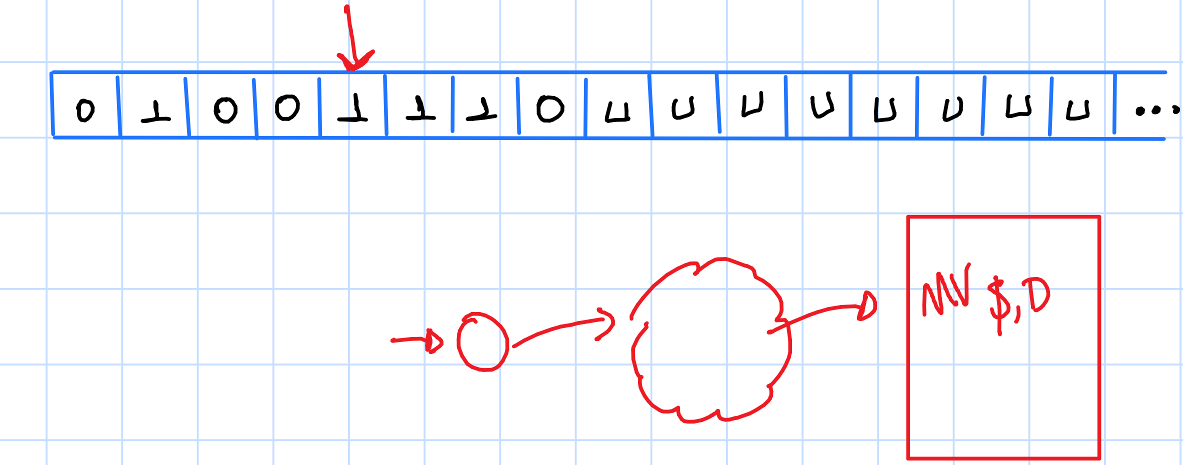
- Podemos assumir que a MT pode detectar o início da fita
- $M_{\$,D}$ poderia gravar: ⊔, ☆, ♡, ...
- $M_{\$,D}$ pode fazer isso em qualquer posição

Desloca a cadeia p/ a direita abrindo espaço p/ escrever $

Preenche com espaço

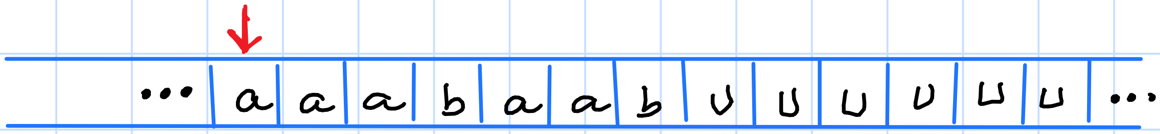
Move p/ esquerda

Podemos fazer $MV_{⊔,E}$



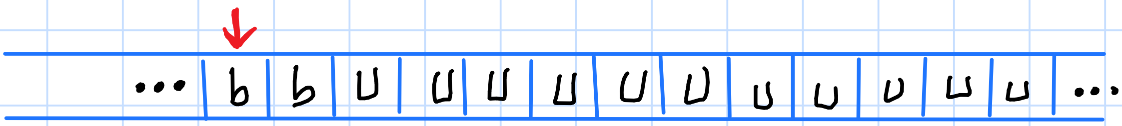

* Podemos retornar a cabeça de leitura para a posição invocada

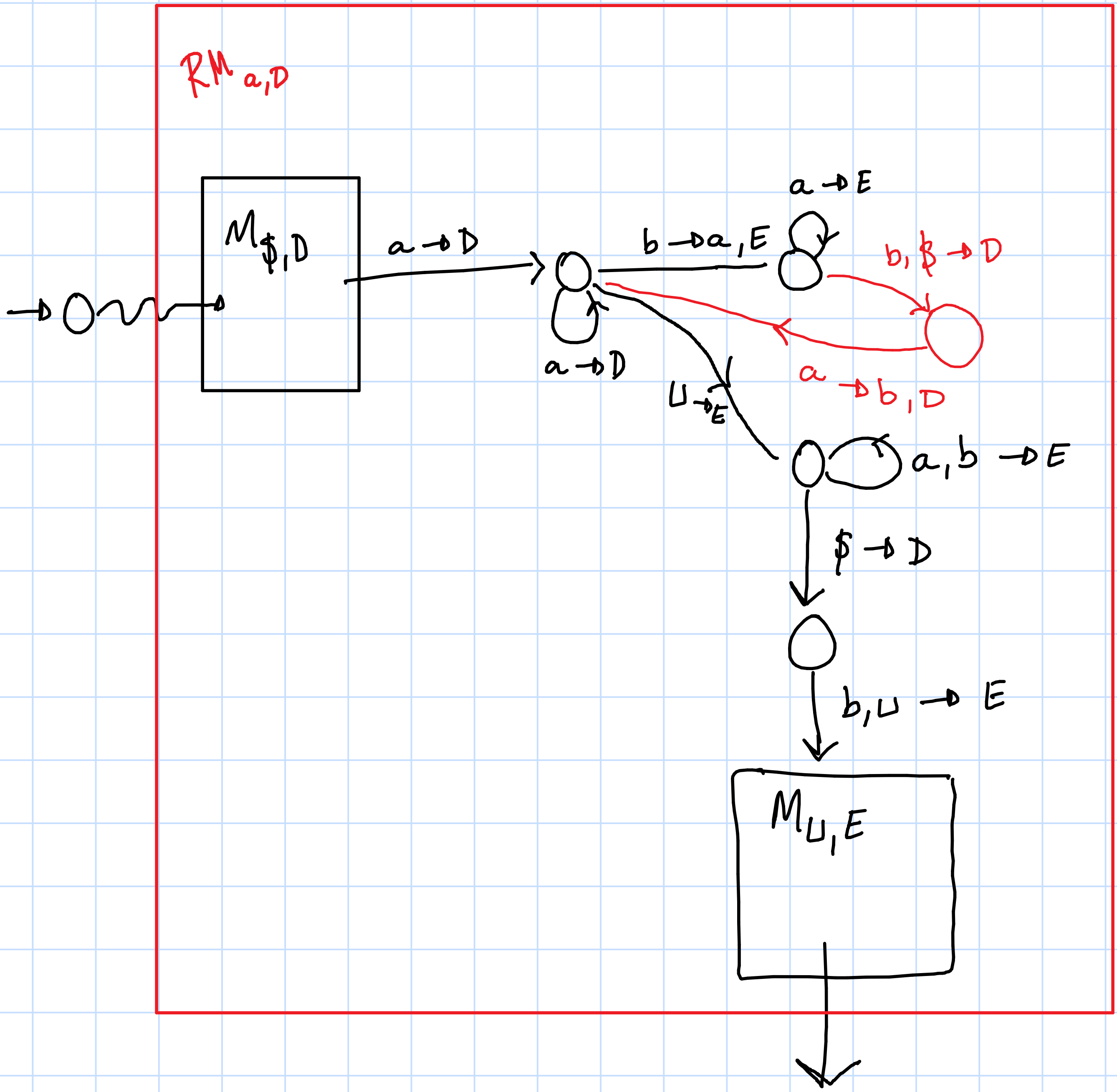
b) Uma máquina $RM_{a,D}$ que apaga caracteres a's à direita da cabeça de leitura e devolve a cabeça de leitura para a posição inicial.



$MV_{\$,D}$

$MV_{\sqcup,E}$




RM a,D
M $,D
a → D
a → D
b → a, E
a → E
b, $ → D
a → b, D
⊔ → E
a, b → E
$ → D
b, ⊔ → E
M ⊔, E

(c) MT que recebe um número em binário e soma 1

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ... |

$MV_{\$,D}$ espaço p/ um possível bit carry

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $ | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ... |

ADD1

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $ | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ... |

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $ | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ... |

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $ | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ... |

⋮

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $ | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ... |

como não precisamo desse espaço p/ carry, podemos deslocar os símbolos p/ a esquerda

(d) MT que copia o trecho à direita no fim da fita

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ⊔ | ... |

# Variações da MT

Existem variações da definição de MT que vimos.

* Fita infinita em ambos os extremos.
* Cabeça fixa.
* Multiplas fitas.
* Não deterministica.

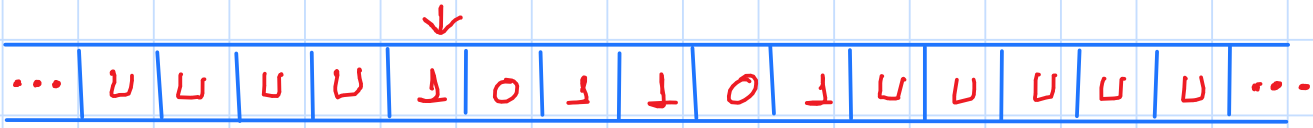
$\boxed{\text{Todas equivalentes}}$

↳ reconhecem o mesmo conjunto de linguagens

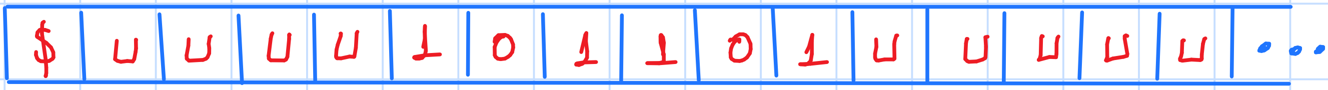
Nota

* adiciona abstração.
* Facilita programação.

# Fita Infinita de ambos os lados




Podemos simular isso com a nossa MT simplesmente movendo a cadeia de entrada para a direita da fita, deixando um espaço suficientemente grande até borda esquerda.

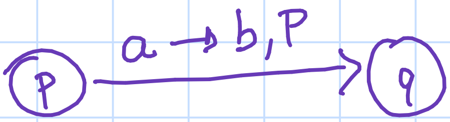


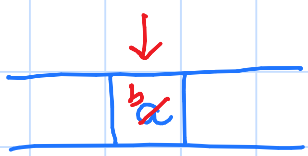

## Cabeça fixa

Nesta versão, além de mover p/ a esquerda e a direita, a cabeça tbm pode ficar parada.

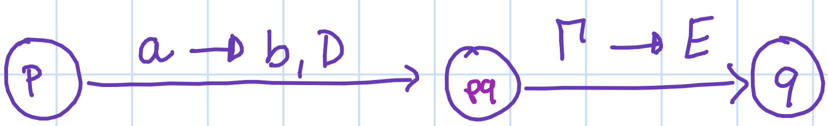
$$\delta : Q \times \Gamma \to Q \times \Gamma \times \{E, D, P\}$$

parada

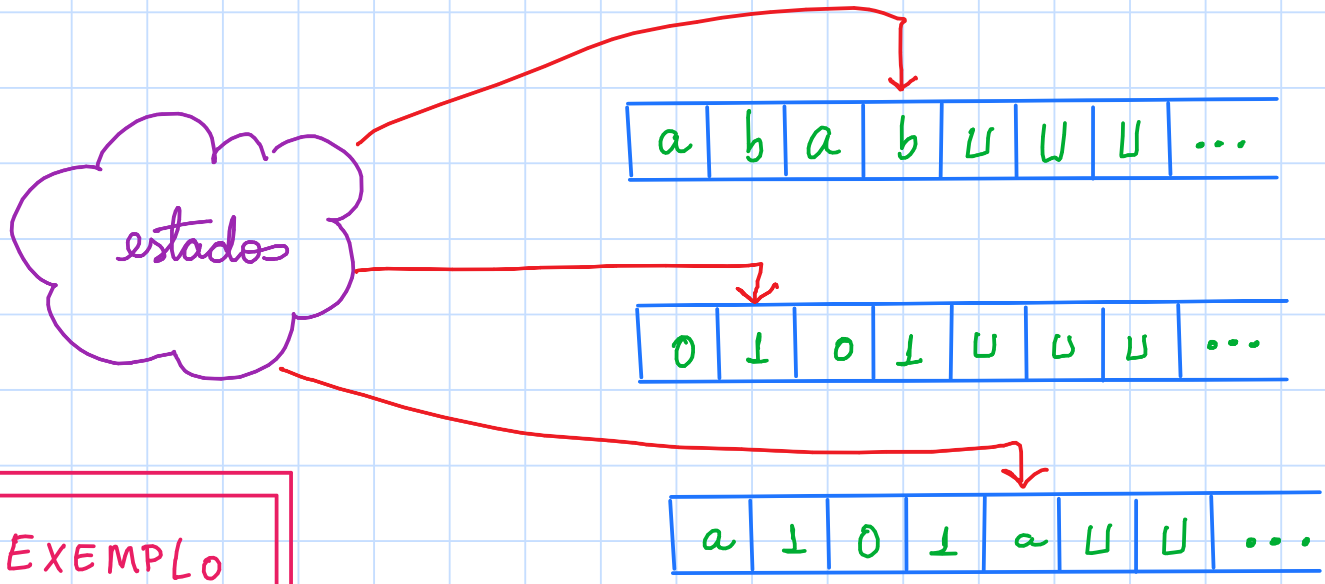



Podemos simular na nossa MT da seguinte forma:

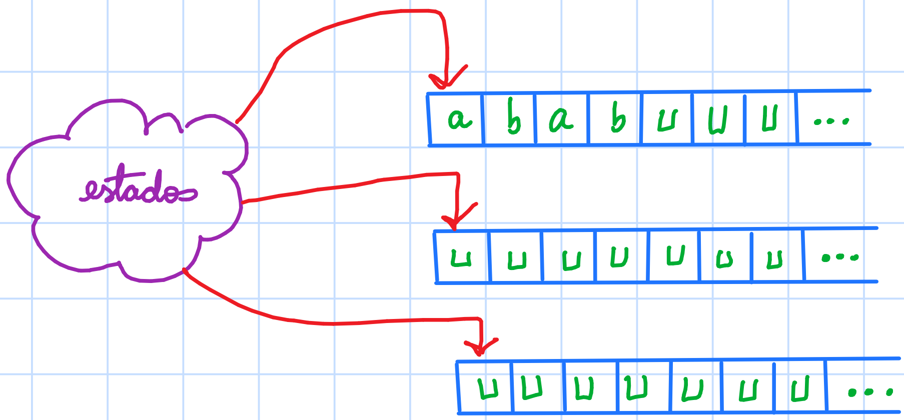


# Multiplas Fitas




EXEMPLO
com K=3

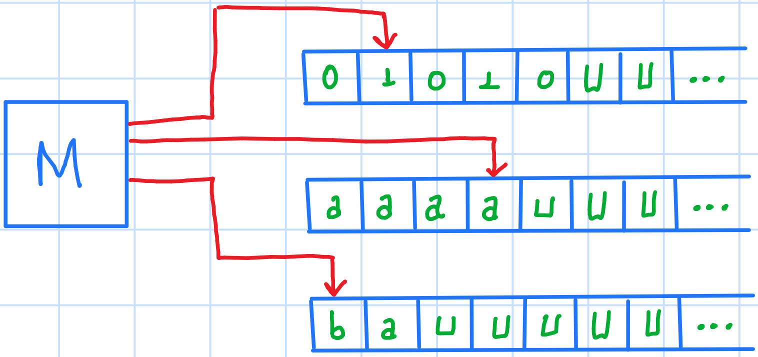
$$Q \xrightarrow{b\ 1\ a \to a\ 0\ Z,\ EED} T$$

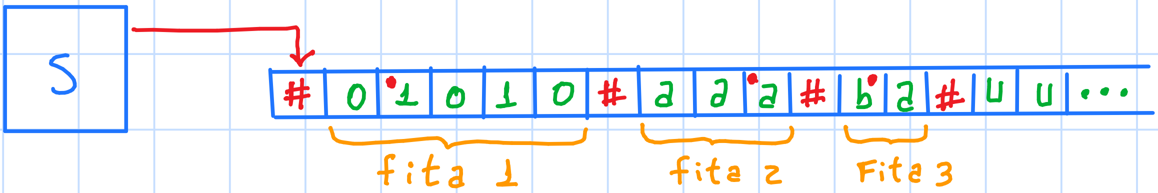


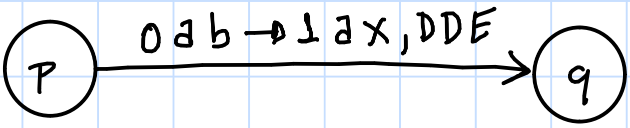

Obs: inicialmente a entrada está na primeira fita e as outras começam em branco.

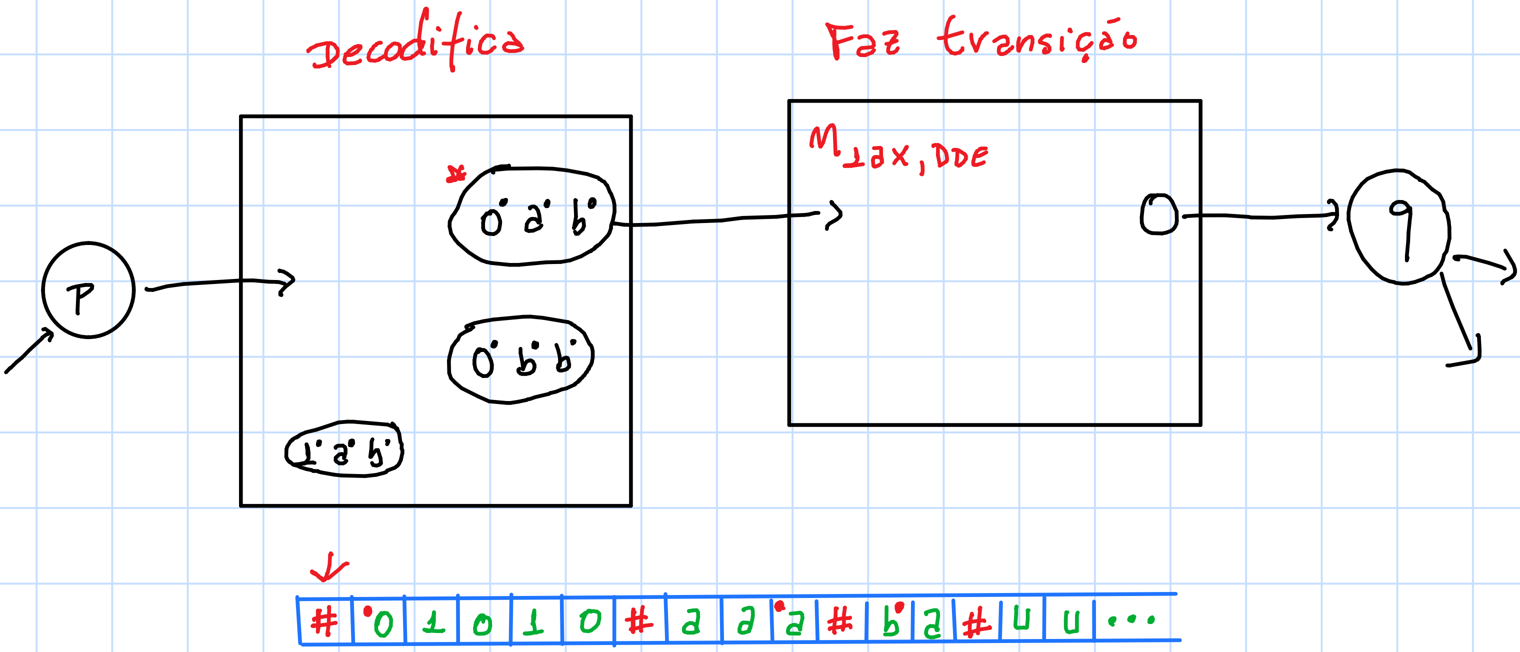
Teorema: Toda MT com multiplas fitas possui uma MT monofita equivalente.

# Ideia de prova



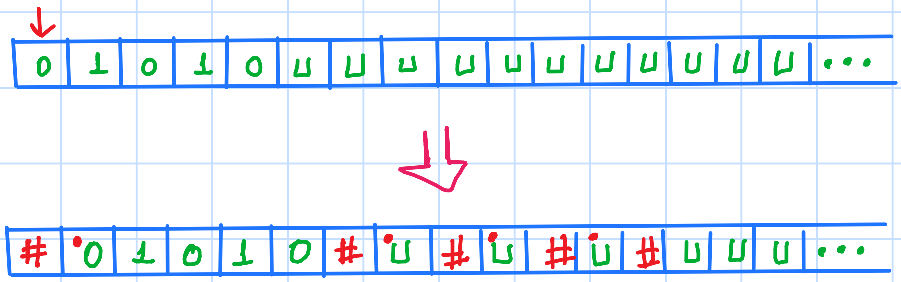



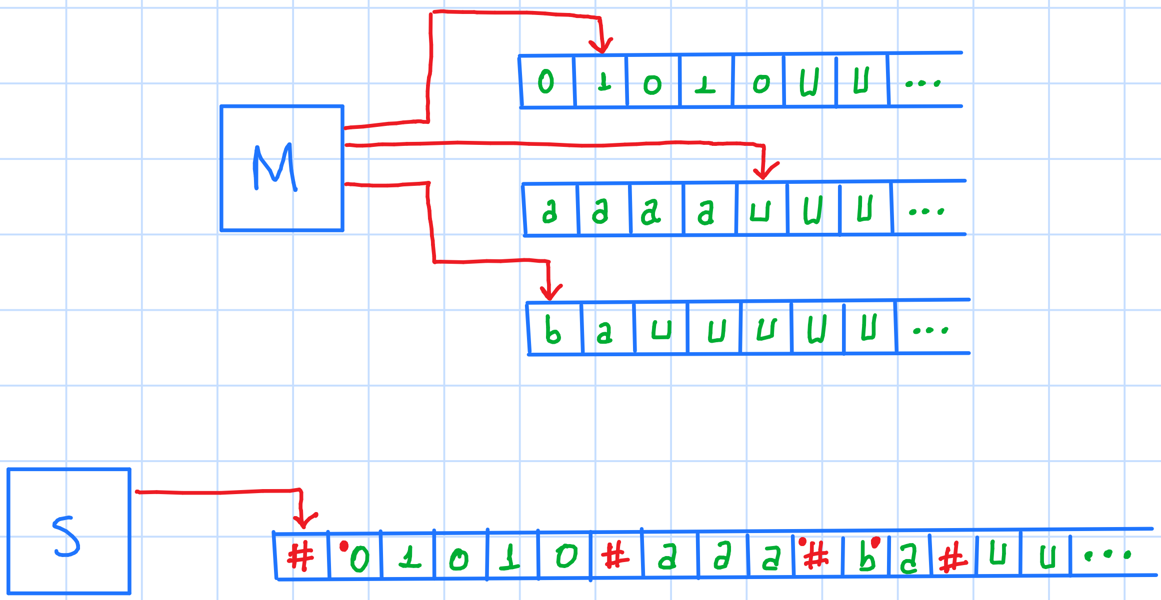



# Comentários finais

## 1. Inicialização



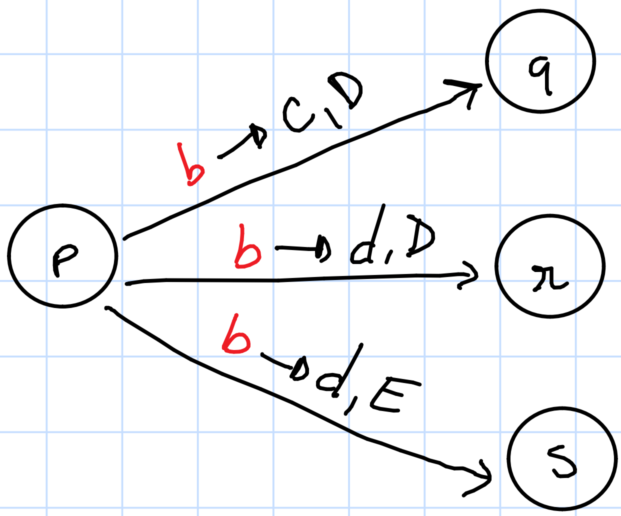

## 2. Quando a "cabeça de leitura virtual" se mover sobre # é necessário deslocar a cadeia na fita.

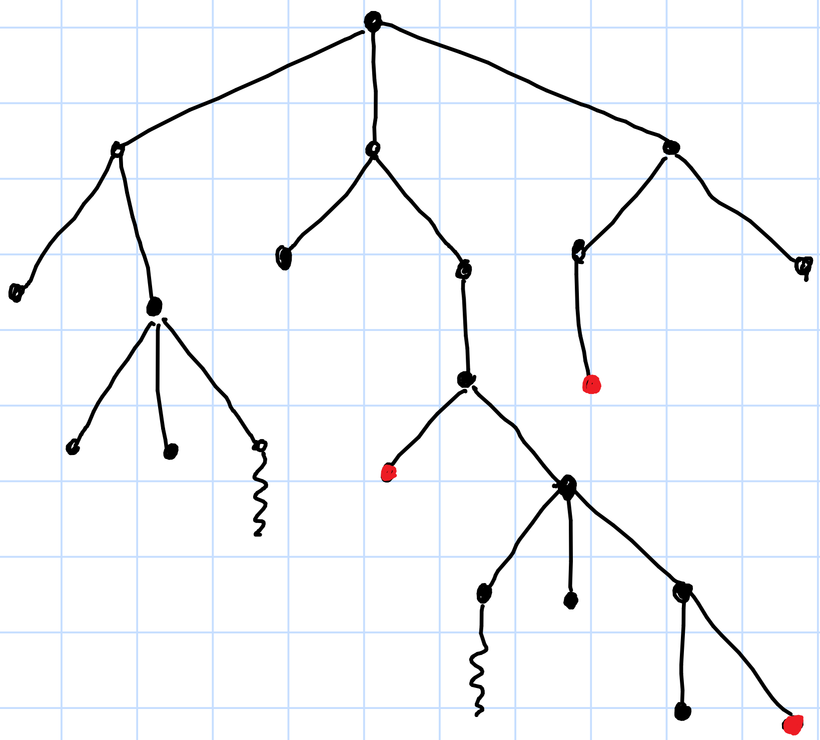



**Corolário** Uma linguagem é Turing-reconhecível se e somente se alguma MT com múltiplas fitas a reconhece.

# Máquina de Turing não Determinística.

$$\delta : Q \times \Gamma \to \mathcal{P}(Q \times \Gamma \times \{E, D\})$$



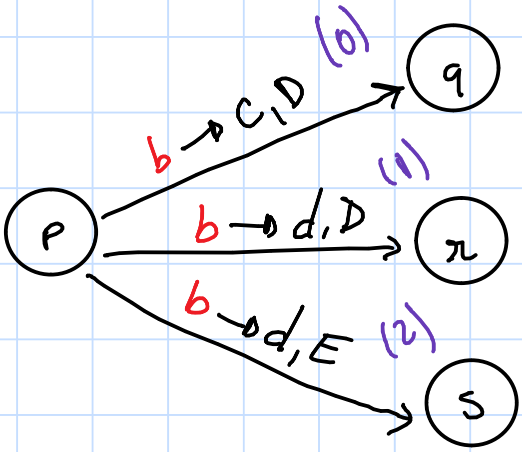



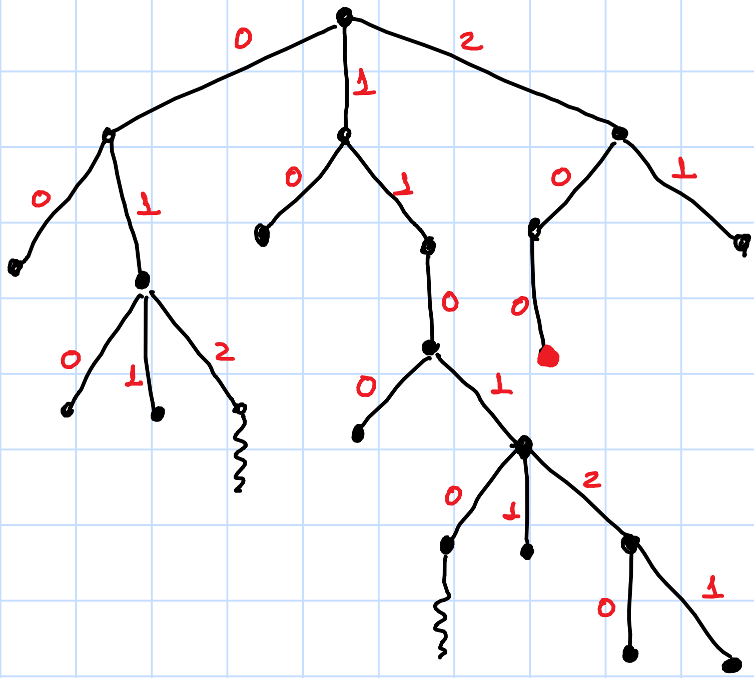
aceita a cadeia se um ramo de computação leva a aceitação da cadeia.

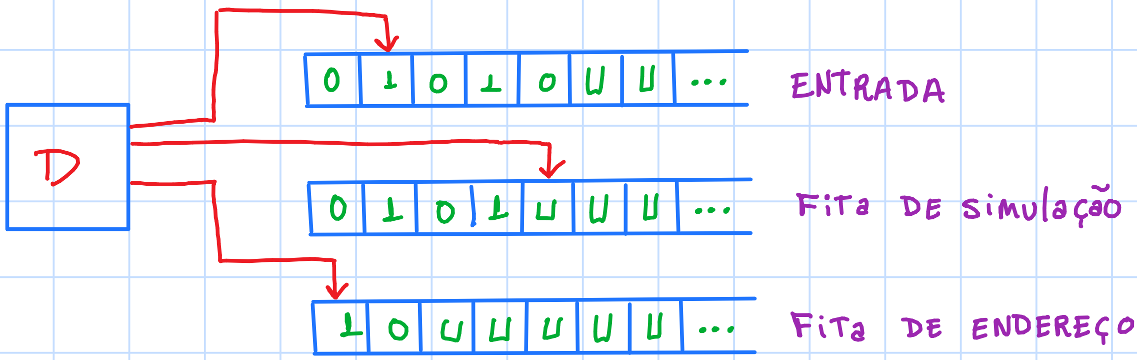
**Teorema** Toda máquina de Turing $\bar{n}$ determinística possui uma máquina de Turing determinística equivalente.

**Ideia da Prova**

$$\delta(p,b) = \{ \overset{0}{(q,c,D)}, \overset{1}{(r,d,D)}, \overset{2}{(s,d,E)} \}$$



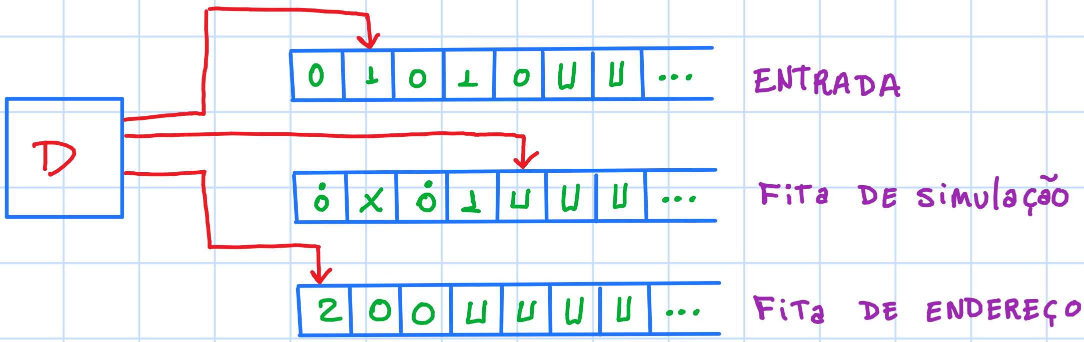



$$\delta(p,b) = \{ \overset{0}{(q,c,D)}, \overset{1}{(r,d,D)}, \overset{2}{(s,d,E)} \}$$

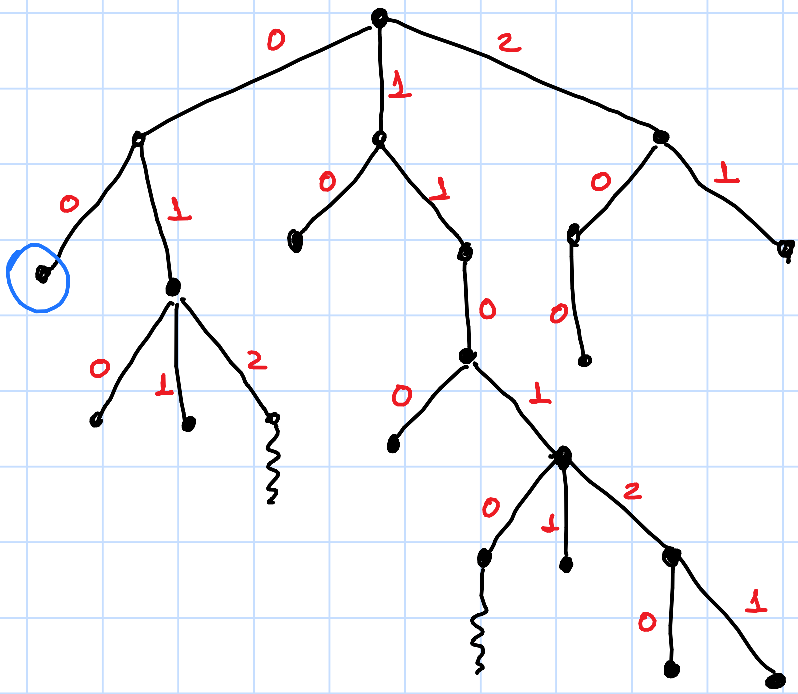
$$b = \mathrm{Max}\ \{ |\delta(r,x)| : r \in Q \text{ e } x \in \Gamma \}$$

Vamos gerar números na base b!

b = 3




20
21
2 22
0 000
1 001
2 002
0 010
1 011
1 2 012




b = 3

20
21
22
0 000
1 0010
2 002
0 010
1 011
2 012

**corolário** uma linguagem é Turing-Reconhecível se e somente se alguma MT não determinística a reconhece.

**corolário** uma linguagem é Turing-Decidível se e somente se alguma MT ñ determinística a decide.