ACH2043
INTRODUÇÃO À TEORIA DA COMPUTAÇÃO

Aula 13

Cap 3.2 – Variantes de MT

Profa. Ariane Machado Lima
ariane.machado@usp.br

Máquinas de Turing – Definição formal

Uma ***máquina de Turing*** é uma 7-upla, $(Q, \Sigma, \Gamma, \delta, q_0, q_{\text{aceita}}, q_{\text{rejeita}})$, onde Q, Σ, Γ são todos conjuntos finitos e

1. Q é o conjunto de estados,
2. Σ é o alfabeto de entrada sem o ***símbolo em branco*** $\sqcup$,
3. Γ é o alfabeto de fita, onde $\sqcup \in \Gamma$ e $\Sigma \subseteq \Gamma$,
4. $\delta: Q \times \Gamma \longrightarrow Q \times \Gamma \times \{\text{E}, \text{D}\}$ é a função de transição,
5. $q_0 \in Q$ é o estado inicial,
6. $q_{\text{aceita}} \in Q$ é o estado de aceitação, e
7. $q_{\text{rejeita}} \in Q$ é o estado de rejeição, onde $q_{\text{rejeita}} \neq q_{\text{aceita}}$.

Máquinas de Turing – Definição formal

Uma ***máquina de Turing*** é uma 7-upla, $(Q, \Sigma, \Gamma, \delta, q_0, q_{\text{aceita}}, q_{\text{rejeita}})$, onde Q, Σ, Γ são todos conjuntos finitos e

1. Q é o conjunto de estados,
2. Σ é o alfabeto de entrada sem o ***símbolo em branco*** $\sqcup$,
3. Γ é o alfabeto de fita, onde $\sqcup \in \Gamma$ e $\Sigma \subseteq \Gamma$,
4. $\delta : Q \times \Gamma \longrightarrow Q \times \Gamma \times \{E, D\}$ é a função de transição,
5. $q_0 \in Q$ é o estado inicial,
6. $q_{\text{aceita}} \in Q$ é o estado de aceitação, e
7. $q_{\text{rejeita}} \in Q$ é o estado de rejeição, onde $q_{\text{rejeita}} \neq q_{\text{aceita}}$.

$\delta : Q' \times \Gamma \longrightarrow Q \times \Gamma \times \{E, D\}$, onde Q' é Q sem q_{aceita} e q_{rejeita}

3.2 – Variantes de Máquinas de Turing

Variantes de Máquinas de Turing

Máquina de Turing é um modelo robusto: ela e suas variações reconhecem a mesma classe de linguagens

Máquinas de Turing Multifita

- K fitas
- Cada fita tem sua própria cabeça para leitura e escrita
- Inicialmente, a cadeia de entrada fica na fita 1, e as demais fitas com branco

$$\delta : Q \times \Gamma^k \longrightarrow Q \times \Gamma^k \times \{\mathrm{E}, \mathrm{D}, \mathrm{P}\}^k$$

$$\delta(q_i, a_1, \ldots, a_k) = (q_j, b_1, \ldots, b_k, \mathrm{E}, \mathrm{D}, \ldots, \mathrm{E})$$

TEOREMA 3.13

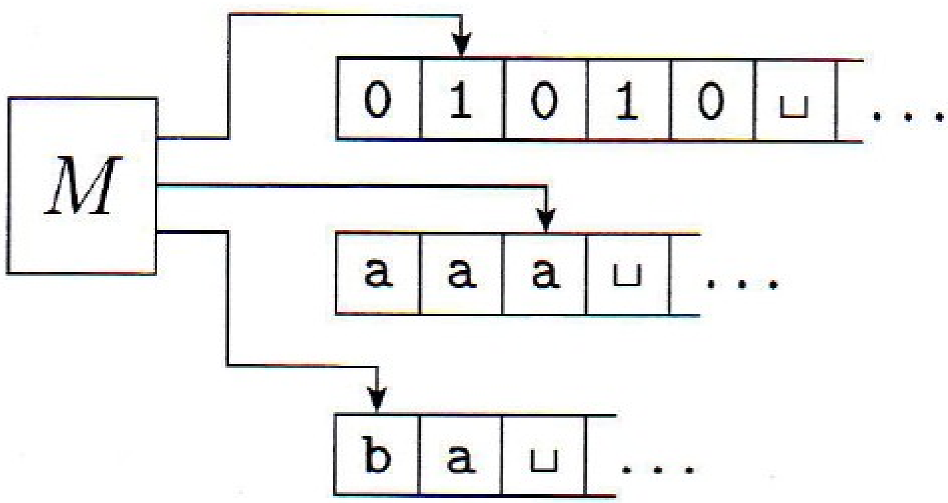
Toda máquina de Turing multifita tem uma máquina de Turing de uma única fita que lhe é equivalente.

TEOREMA 3.13

Toda máquina de Turing multifita tem uma máquina de Turing de uma única fita que lhe é equivalente.

Seja S uma MT de fita única e M uma MT de k fitas.

S pode simular M:

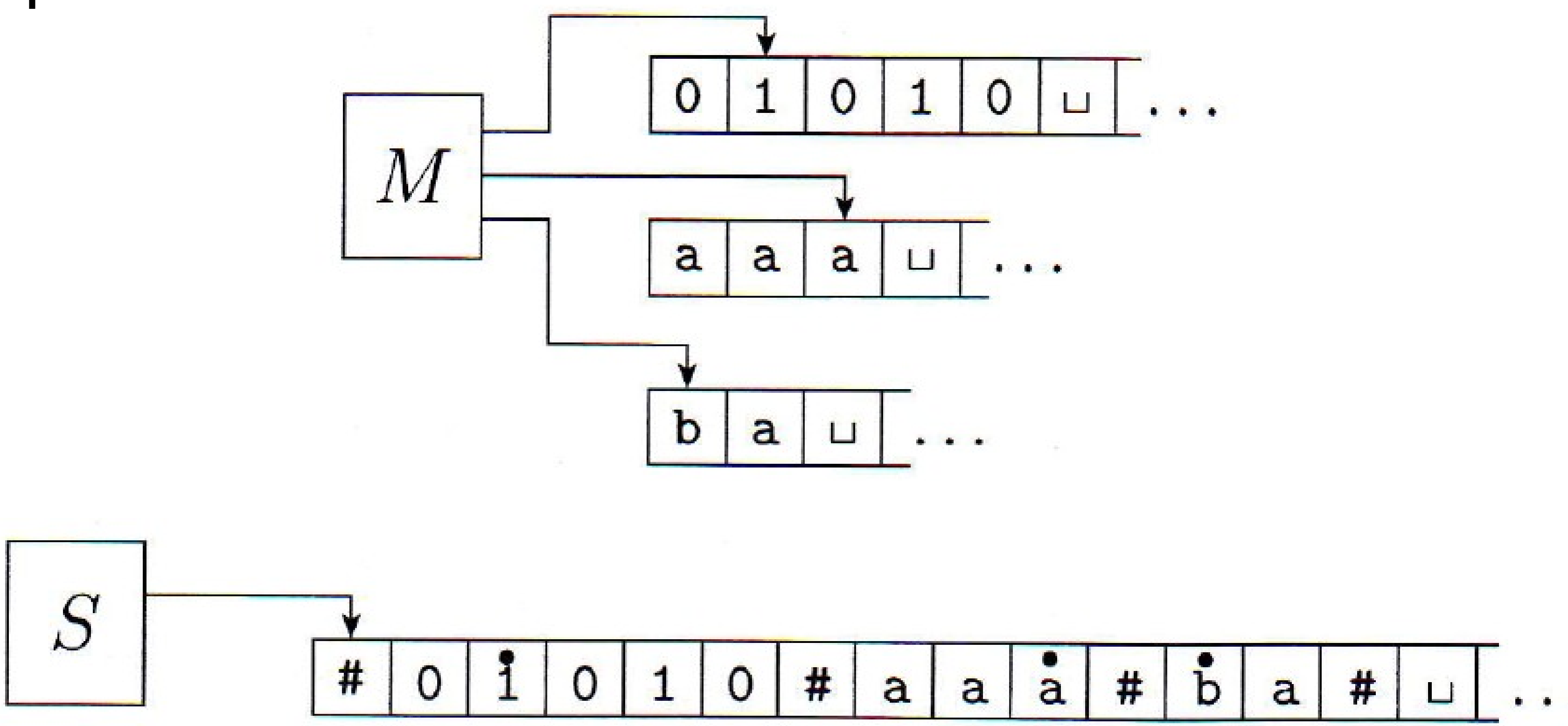


TEOREMA 3.13

Toda máquina de Turing multifita tem uma máquina de Turing de uma única fita que lhe é equivalente.

Seja S uma MT de fita única e M uma MT de k fitas.

S pode simular M:



Funcionamento de S:

- Preparação da fita: (ex: $w = w_1 \cdots w_n$)

Funcionamento de S:

- Preparação da fita: (ex: $w = w_1 \cdots w_n$)

$$\#\dot{w}_1 w_2 \cdots w_n \#\dot{\sqcup}\#\dot{\sqcup}\# \cdots \#$$

Funcionamento de S:

- Preparação da fita: (ex: $w = w_1 \cdots w_n$)

$$\#\dot{w}_1 w_2 \cdots w_n \#\dot{\sqcup}\#\dot{\sqcup}\# \cdots \#$$

- Leitura dos símbolos atuais:

Funcionamento de S:

- Preparação da fita: (ex: $w = w_1 \cdots w_n$)

$$\#\dot{w}_1 w_2 \cdots w_n \#\dot{\sqcup}\#\dot{\sqcup}\# \cdots \#$$

- Leitura dos símbolos atuais: percorre a fita lendo os símbolos com ponto em cima (até o (k+1)-ésimo #)

Funcionamento de S:

- Preparação da fita: (ex: $w = w_1 \cdots w_n$)

$$\# \dot{w}_1 w_2 \cdots w_n \# \dot{\sqcup} \# \dot{\sqcup} \# \cdots \#$$

- Leitura dos símbolos atuais: percorre a fita lendo os símbolos com ponto em cima (até o (k+1)-ésimo #)

  Atualização das cabeças:

Funcionamento de S:

- Preparação da fita: (ex: $w = w_1 \cdots w_n$)

$$\# \dot{w}_1 w_2 \cdots w_n \# \dot{\sqcup} \# \dot{\sqcup} \# \cdots \#$$

- Leitura dos símbolos atuais: percorre a fita lendo os símbolos com ponto em cima (até o (k+1)-ésimo #)

  Atualização das cabeças: percorre a fita fazendo as atualizações conforme a função de transição (tirando e colocando pontos para atualizar as cabeças de fitas), (até o (k+1)-ésimo #)

Funcionamento de S:

- Preparação da fita: (ex: $w = w_1 \cdots w_n$)

$$\# \dot{w}_1 w_2 \cdots w_n \# \dot{\sqcup} \# \dot{\sqcup} \# \cdots \#$$

- Leitura dos símbolos atuais: percorre a fita lendo os símbolos com ponto em cima (até o (k+1)-ésimo #)

  Atualização das cabeças: percorre a fita fazendo as atualizações conforme a função de transição (tirando e colocando pontos para atualizar as cabeças de fitas), (até o (k+1)-ésimo #)

- Se uma cabeça de fita vai para um "#":

Funcionamento de S:

- Preparação da fita: (ex: $w = w_1 \cdots w_n$)

$$\#\dot{w}_1 w_2 \cdots w_n \#\dot{\sqcup}\#\dot{\sqcup}\# \cdots \#$$

- Leitura dos símbolos atuais: percorre a fita lendo os símbolos com ponto em cima (até o (k+1)-ésimo #)

  Atualização das cabeças: percorre a fita fazendo as atualizações conforme a função de transição (tirando e colocando pontos para atualizar as cabeças de fitas), (até o (k+1)-ésimo #)
- Se uma cabeça de fita vai para um "#": desloca o conteúdo da fita para a direita e coloca o símbolo no lugar daquele "#"

COROLÁRIO 3.15

Uma linguagem é Turing-reconhecível se e somente se alguma máquina de Turing multifita a reconhece.

COROLÁRIO 3.15

Uma linguagem é Turing-reconhecível se e somente se alguma máquina de Turing multifita a reconhece.

Prova:

Linguagem L é TR => MTM reconhece L:

- L é TR => existe uma MT de fita única que a reconhece => existe uma MT multifita que a reconhece (pois fita única é um caso especial de multifita)

MTM reconhece L => Linguagem L é TR:

- MTM reconhece L => uma MT fita única a reconhece (pelo teorema) => L é TR

Máquinas de Turing Não-Determinísticas

Máquinas de Turing Não-Determinísticas

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

TEOREMA 3.16

Toda máquina de Turing não-determinística tem uma máquina de Turing determinística que lhe é equivalente.

TEOREMA 3.16

Toda máquina de Turing não-determinística tem uma máquina de Turing determinística que lhe é equivalente.

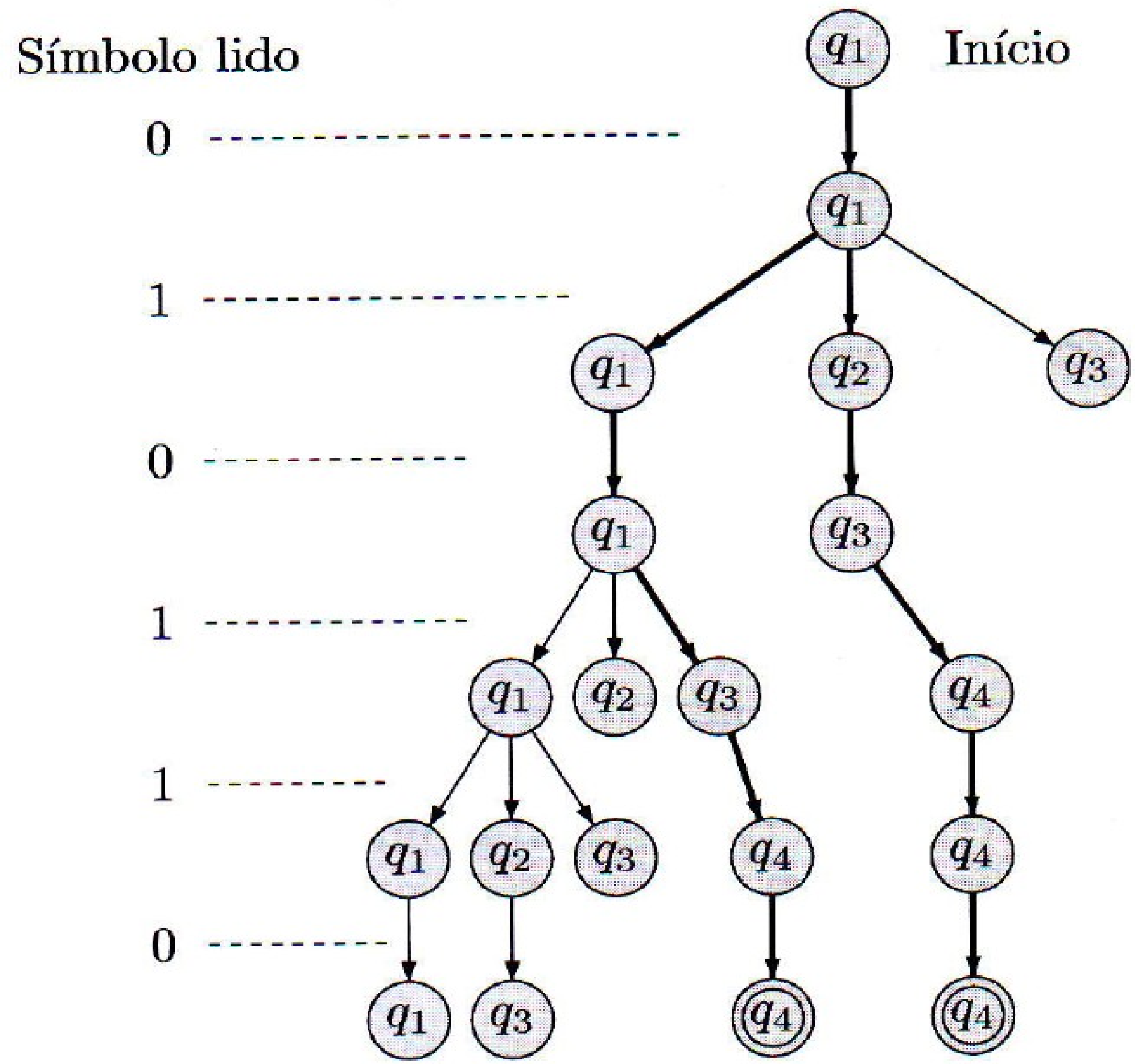
Ideia da prova:

Usar para MT determinística D para simular todas as possíveis computações de uma MT não determinística N

Computação de N representada por uma árvore (filhos de um nó são as possibilidades de transição)

Cada caminho (a partir da raiz) é uma computação possível

Lembram da árvore para AFNDs?



TEOREMA 3.16

Toda máquina de Turing não-determinística tem uma máquina de Turing determinística que lhe é equivalente.

Ideia da prova:

Usar para MT determinística D para simular todas as possíveis computações de uma MT não determinística N

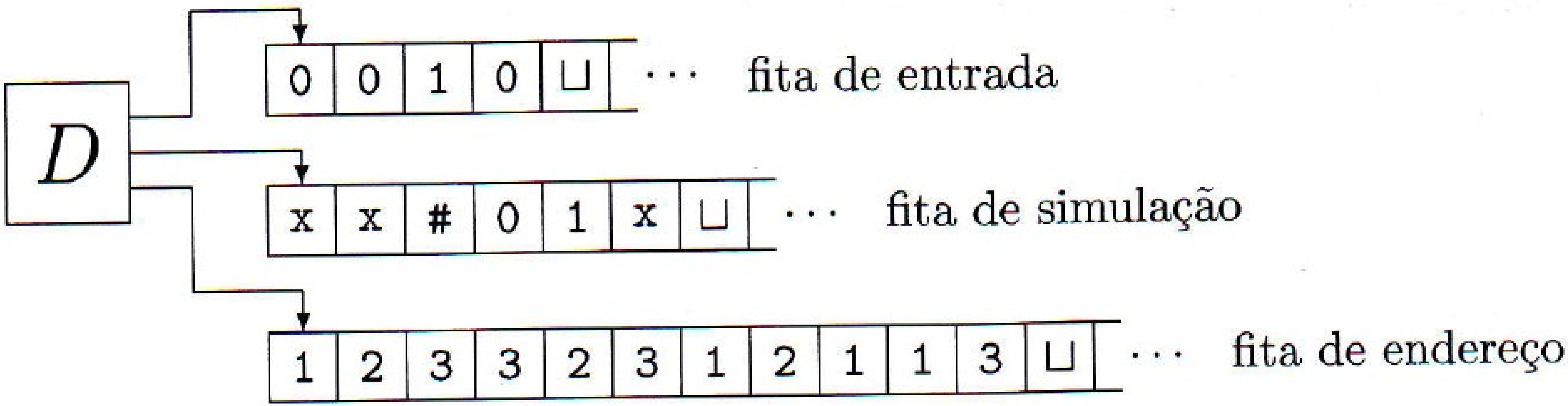
Computação de N representada por uma árvore (filhos de um nó são as possibilidades de transição)

Cada caminho (a partir da raiz) é uma computação possível

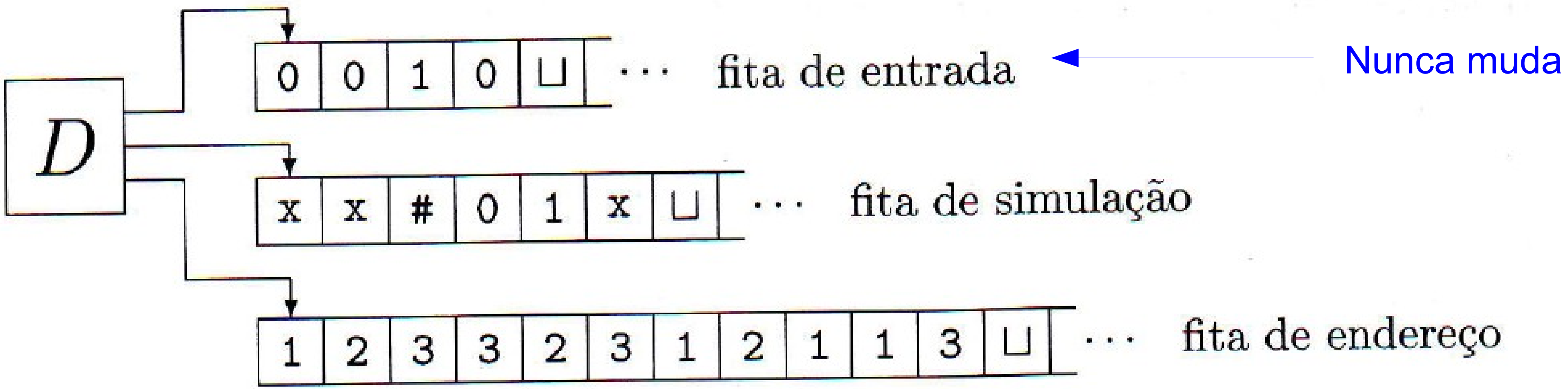
Cada nó dessa árvore terá um endereço que indica qual filho seguir a partir da raiz (ex: 314)

D irá percorrer essa árvore através de busca em largura (para impedir que caia em um ramo infinito)

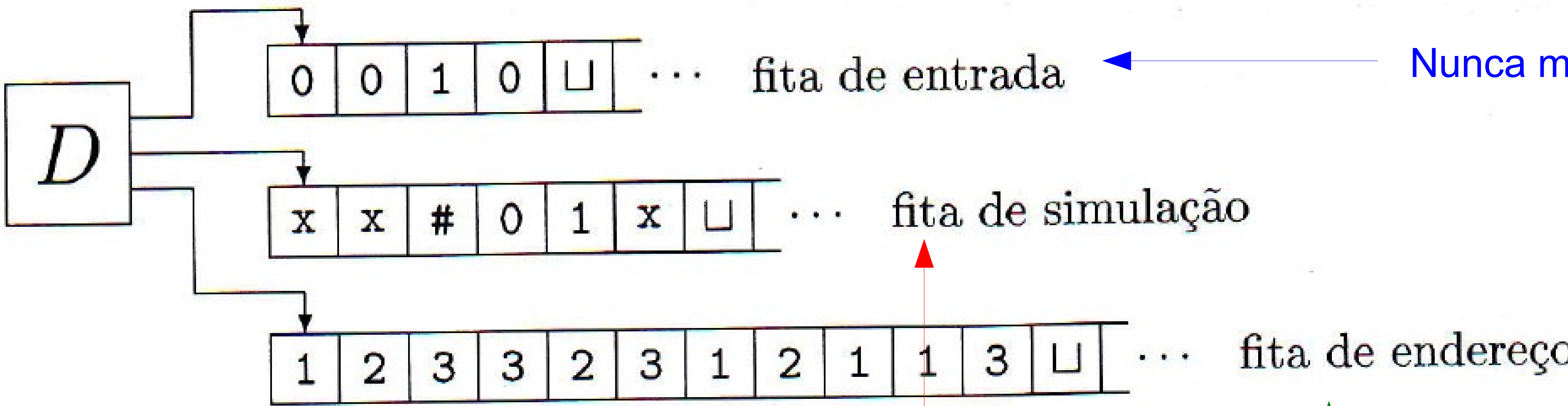
Prova


D
0 0 1 0 ⊔ ...
fita de entrada
x x # 0 1 x ⊔ ...
fita de simulação
1 2 3 3 2 3 1 2 1 1 3 ⊔ ...
fita de endereço

Prova


D
0 0 1 0 ⊔ ···
fita de entrada
Nunca muda
x x # 0 1 x ⊔ ···
fita de simulação
1 2 3 3 2 3 1 2 1 1 3 ⊔ ···
fita de endereço

Prova

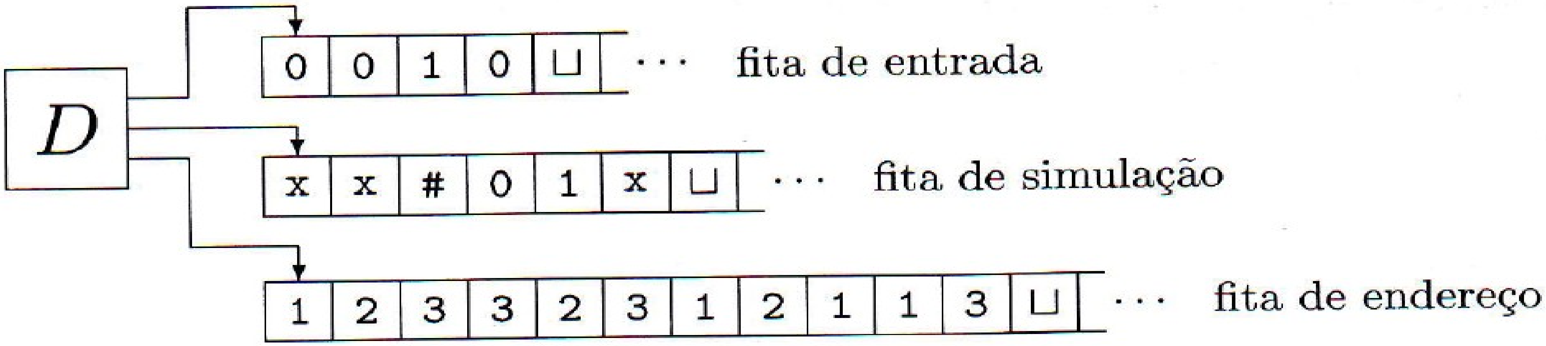


Nunca muda

Usada para simular cada ramo separadamente

Usada para indicar em qual computação (caminho) estou
Possui uma cadeia em Σ_b^* , $\Sigma_b = \{1, 2, \ldots, b\}$, sendo b o maior número de filhos de um nó (ie, o maior número de transições alternativas de um estado)

Prova



1. Inicialmente, a fita 1 contém a entrada w e as fitas 2 e 3 estão vazias.
2. Copie a fita 1 para a fita 2.
3. Use a fita 2 para simular N com a entrada w sobre um ramo de sua computação não-determinística. Antes de cada passo de N, consulte o próximo símbolo na fita 3 para determinar qual escolha fazer entre aquelas permitidas pela função de transição de N. Se não restam mais símbolos na fita 3 ou se essa escolha não-determinística for inválida, aborte esse ramo indo para o estágio 4. Também vá para o estágio 4 se uma configuração de rejeição for encontrada. Se uma configuração de aceitação for encontrada, *aceite* a entrada.
4. Substitua a cadeia na fita 3 pela próxima cadeia na ordem lexicográfica. Simule o próximo ramo da computação de N indo para o estágio 2.

COROLÁRIO 3.18

Uma linguagem é Turing-reconhecível se e somente se alguma máquina de Turing não-determinística a reconhece.