ACH2043
INTRODUÇÃO À TEORIA DA COMPUTAÇÃO

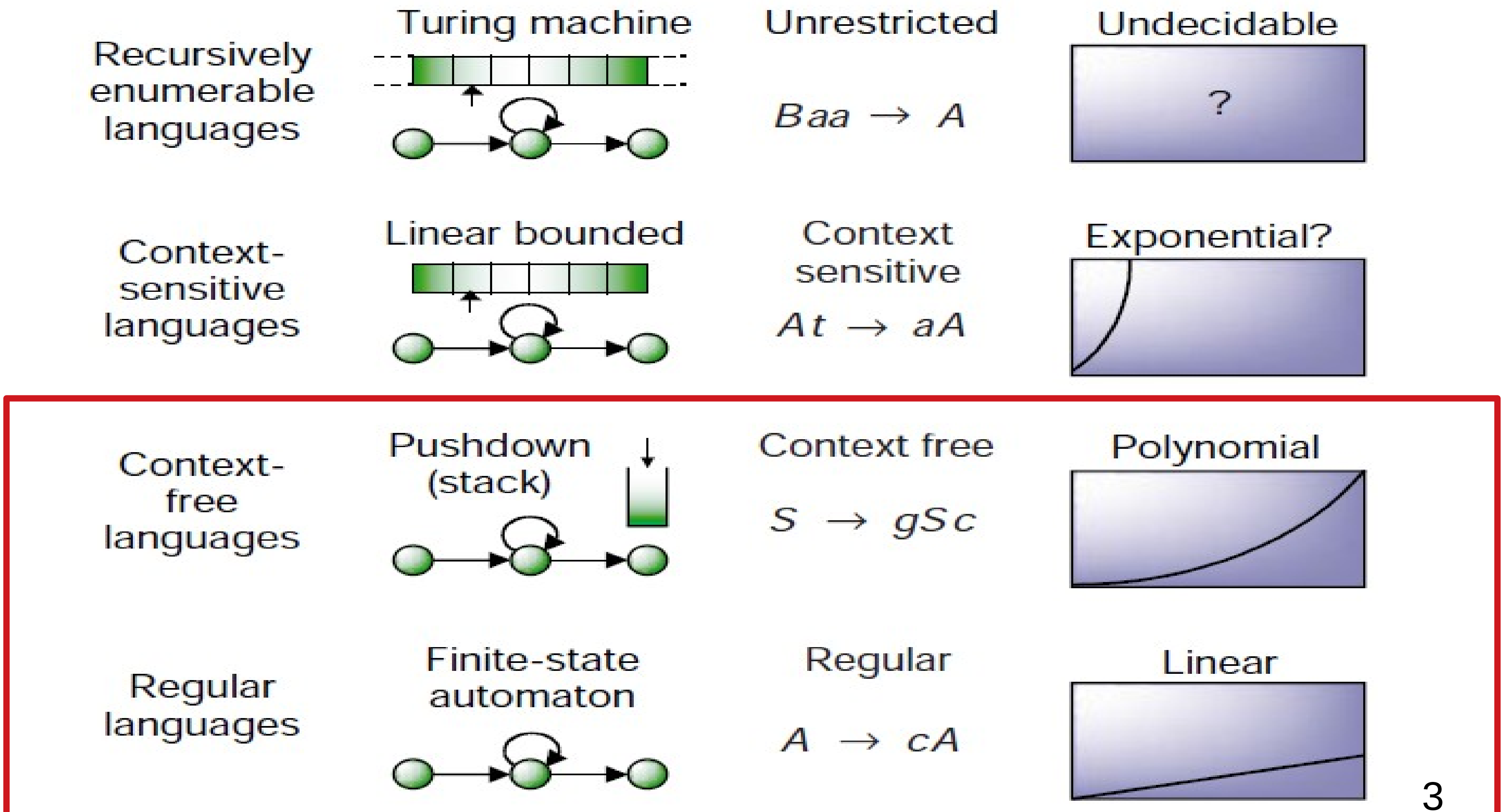
Aula 16

Cap 3 – Máquinas de Turing

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

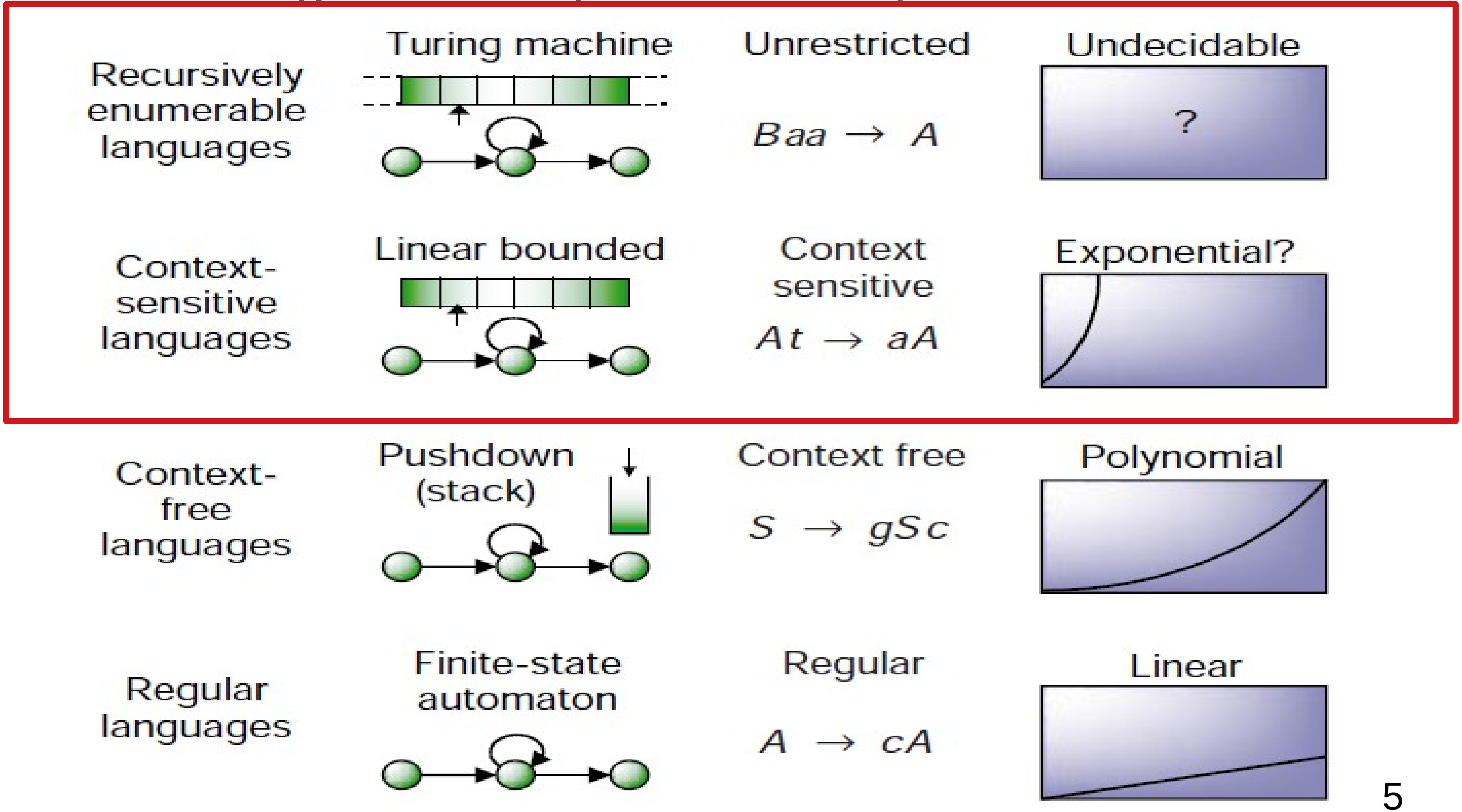
Aulas anteriores

Linguagens, modelos computacionais (dispositivos, gramáticas) e suas complexidades



Aulas de hoje

Linguagens, modelos computacionais (dispositivos, gramáticas) e suas complexidades



Cap. 3 - A tese de Church-Turing

3.1 – Máquinas de Turing

3.2 – Variantes da Máquinas de Turing

3.3 – A Definição de Algoritmo

3.1 - Máquinas de Turing

- Autômatos como modelos de computação:
  - AF: memória pequena
  - AP: memória ilimitada mas utilizável apenas em sistema LIFO (last in, first out) de leitura

3.1 - Máquinas de Turing

- Autômatos como modelos de computação:
  - AF: memória pequena
  - AP: memória ilimitada mas utilizável apenas em sistema LIFO (last in, first out) de leitura
  - Máquinas de Turing: memória ilimitada e irrestrita

3.1 - Máquinas de Turing

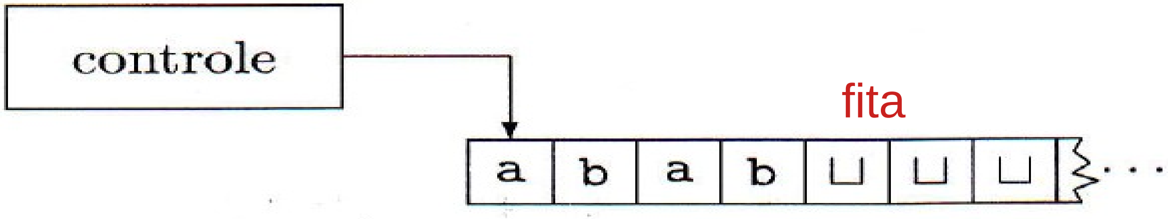
- Propostas por Alan Turing em 1936
  - Memória ilimitada e irrestrita
  - Modelo de um computador real (possibilidades e limitações)

Será nossa base para estudarmos
- computabilidade (o que é possível fazer com o computador)
- complexidade (em que grandeza de tempo)

Alan Turing

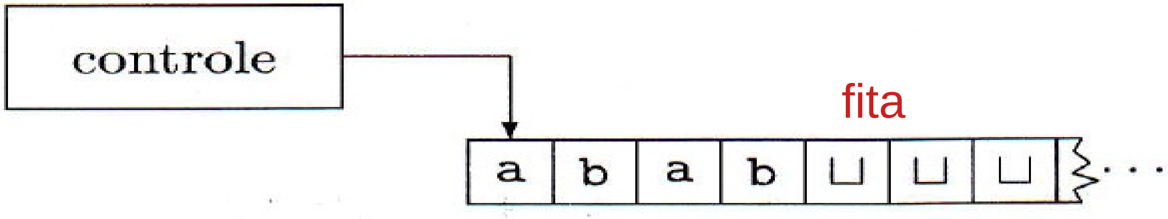
Também dedicou-se a criptografia...

Máquinas de Turing



1. Uma máquina de Turing pode tanto escrever sobre a fita quanto ler a partir dela.
2. A cabeça de leitura-escrita pode mover-se tanto para a esquerda quanto para a direita.
3. A fita é infinita.
4. Os estados especiais para rejeitar e aceitar fazem efeito imediatamente.

Máquinas de Turing



1. Uma máquina de Turing pode tanto escrever sobre a fita quanto ler a partir dela.
2. A cabeça de leitura-escrita pode mover-se tanto para a esquerda quanto para a direita.
3. A fita é infinita. (à direita)
4. Os estados especiais para rejeitar e aceitar fazem efeito imediatamente.

A cadeia de entrada é inserida na fita para poder ir e voltar sobre ela à vontade.

Máquinas de Turing - Tipos de descrições

1-) Descrição formal: diagrama de estados ou definição matemática (conjuntos, função de transição, etc.)

2-) Descrição de implementação (descrição em língua natural da manipulação da máquina – como escreve/lê da fita)

3-) Descrição de alto-nível (algoritmo - pseudocódigo)

Máquinas de Turing - Tipos de descrições

1-) Descrição formal: diagrama de estados ou definição matemática (conjuntos, função de transição, etc.)

2-) Descrição de implementação (descrição em língua natural da manipulação da máquina – como escreve/lê da fita)

3-) Descrição de alto-nível (algoritmo - pseudocódigo)

Máquinas de Turing – Definição formal

Uma ***máquina de Turing*** é uma 7-upla, $(Q, \Sigma, \Gamma, \delta, q_0, q_{aceita}, q_{rejeita})$, 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. $q_{aceita} \in Q$ é o estado de aceitação, e
7. $q_{rejeita} \in Q$ é o estado de rejeição, onde $q_{rejeita} \neq q_{aceita}$.

Cursor da fita vai para a esquerda ou direita

Máquinas de Turing – Definição formal

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

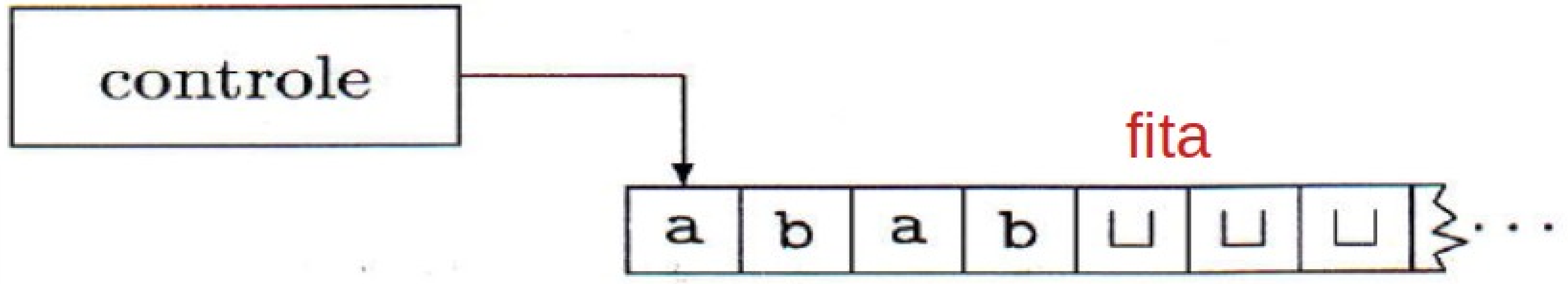
1. Q é o conjunto de estados,
2. Σ é o alfabeto de entrada sem o ***símbolo em branco*** $\sqcup$,
3. Γ é o alfabeto de fita, onde $\sqcup \in \Gamma$ e $\Sigma \subseteq \Gamma$,
4. $\delta: Q \times \Gamma \longrightarrow Q \times \Gamma \times \{E, D\}$ é a função de transição,
5. $q_0 \in Q$ é o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

Mais precisamente:

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

Máquinas de Turing - Funcionamento

- Inicialmente a entrada fica na porção mais à esquerda da fita
- O símbolo em branco marca o fim da entrada
- A máquina começa apontando para a primeira posição da fita
- Se a máquina está na primeira posição e tenta fazer um movimento para a esquerda, permanece no lugar



Máquinas de Turing - Funcionamento

- Inicialmente a entrada fica na porção mais à esquerda da fita
- O símbolo em branco marca o fim da entrada
- A máquina começa apontando para a primeira posição da fita
- Se a máquina está na primeira posição e tenta fazer um movimento para a esquerda, permanece no lugar
- Pára SOMENTE quando entra em um estado de aceitação ou rejeição (se não, NÃO PÁRA NUNCA!)

Máquinas de Turing - Exemplo

B= { w # w | w pertence a {0,1}*}

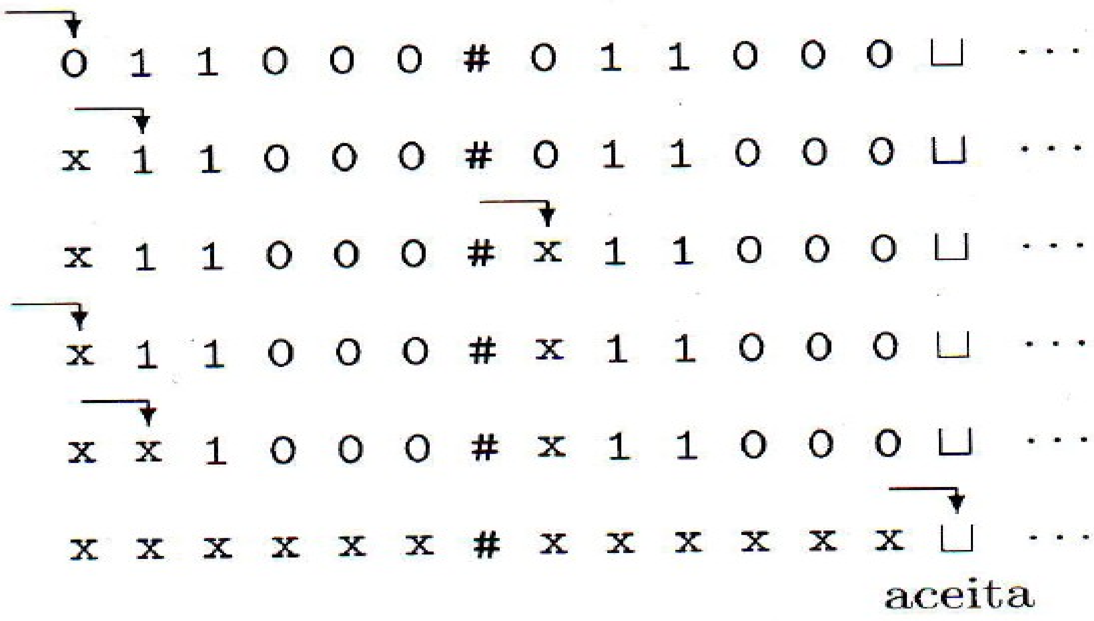
Máquinas de Turing - Exemplo

B= { w # w | w pertence a {0,1}*}

0 1 1 0 0 0 # 0 1 1 0 0 0 ␣ . . .

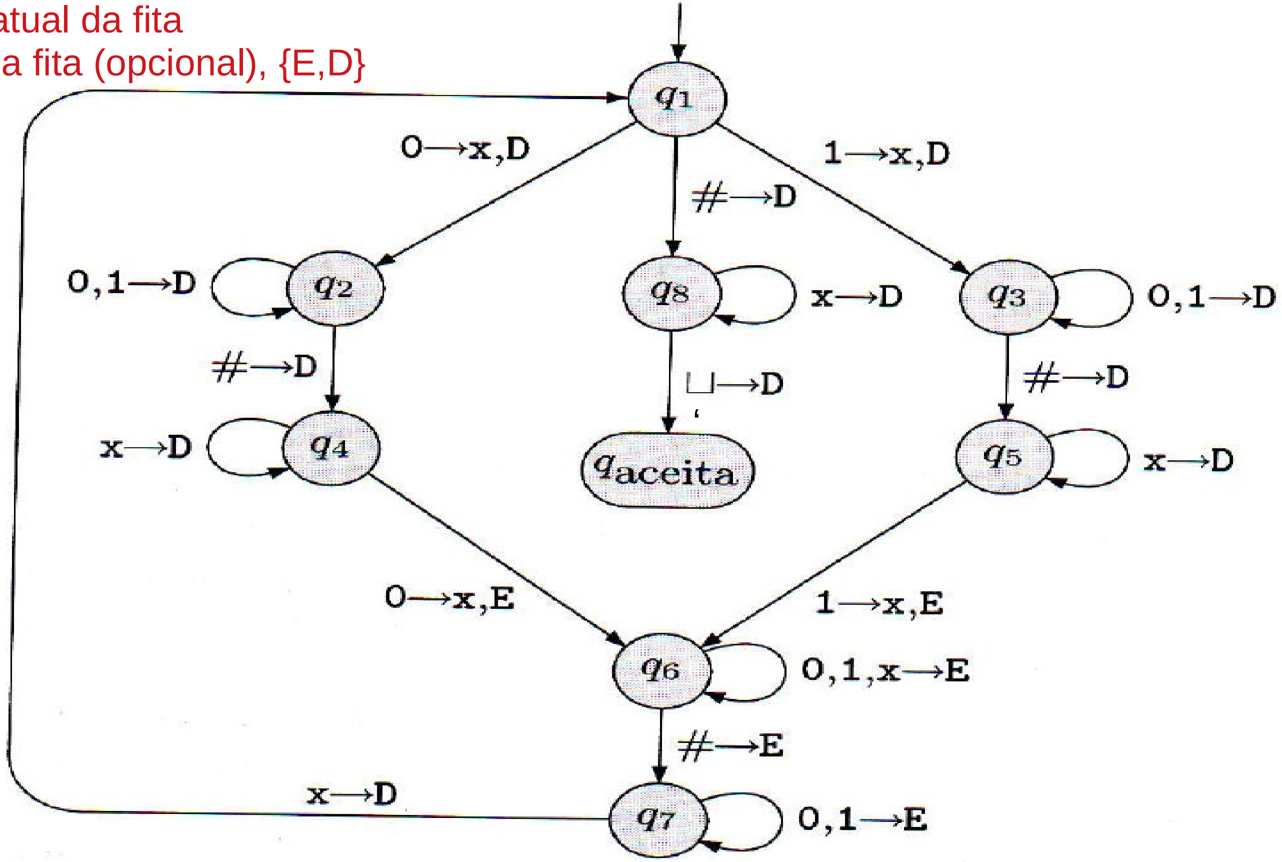
Máquinas de Turing - Exemplo

B= { w # w | w pertence a {0,1}*}

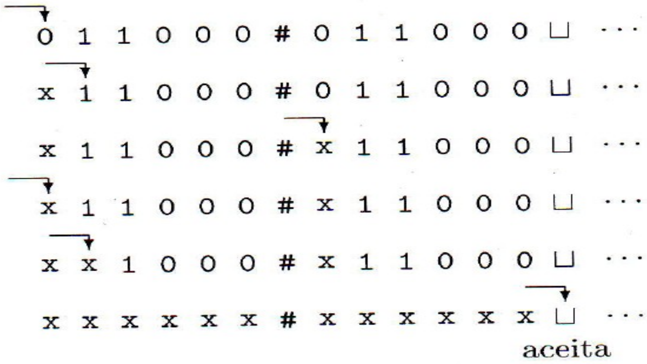


Notação: A → B:
A: lista de possíveis símbolos na posição atual da fita
B: símbolo a ser escrito na posição atual da fita (opcional), {E,D}

Descrição formal



Cada laço tenta casar um símbolo do lado esquerdo do # com um símbolo do lado direito, até que todas as correspondências sejam encontradas



Transições implícitas para $q_{rejeita}$ (indo para a direita, por convenção) quando aparece um símbolo não definido na transição.

Máquinas de Turing - Tipos de descrições

1-) Descrição formal: diagrama de estados ou definição matemática (conjuntos, função de transição, etc.)

2-) Descrição de implementação (descrição em língua natural da manipulação da máquina)

3-) Descrição de alto-nível (algoritmo)

Máquinas de Turing - Tipos de descrições

Nome da máquina (como se fosse o nome da sua função)

código

M = "Sobre a entrada .....

1. ...
2. ....

.... "

Definição da entrada (parâmetros!!!)

Abre e fecha aspas define a máquina

2-) Descrição de implementação (descrição em língua natural da manipulação da máquina)

São como funções booleanas:

```
bool M(...){
  ...
}
```

que dentro do código precisam retornar *true* ou *false* dependendo da entrada

Isto é: *aceitar* ou *rejeitar* a entrada

3-) Descrição de alto-nível (algoritmo)

Máquinas de Turing – Exemplo (descrição de implementação)

B= { w # w | w pertence a {0,1}*}

M_1 = "Sobre a cadeia de entrada w:

1. Faça um 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 contêm, ou se nenhum # for encontrado, *rejeite*. Marque os símbolos à medida que eles são verificados para manter registro de quais símbolos têm correspondência.
2. Quando todos os símbolos à esquerda do # tiverem sido marcados, verifique a existência de algum símbolo remanescente à direita do #. Se resta algum símbolo, *rejeite*; caso contrário, *aceite*."

Máquinas de Turing – Exemplo (descrição de alto-nível)

B= { w # w | w pertence a {0,1}*}

M1 = “Sobre a cadeia de entrada w:

1. Verifique se todo o lado esquerdo de “#” é exatamente igual ao lado direito.
2. Se forem iguais *aceite*, senão *rejeite*.”

Vídeos recomendados

Computerphile: https://www.youtube.com/watch?v=dNRDvLACg5Q

Uma construção moderna de uma máquinas de Turing:

https://www.youtube.com/watch?v=E3keLeMwfHY

Máquinas de Turing -

Formalização do funcionamento

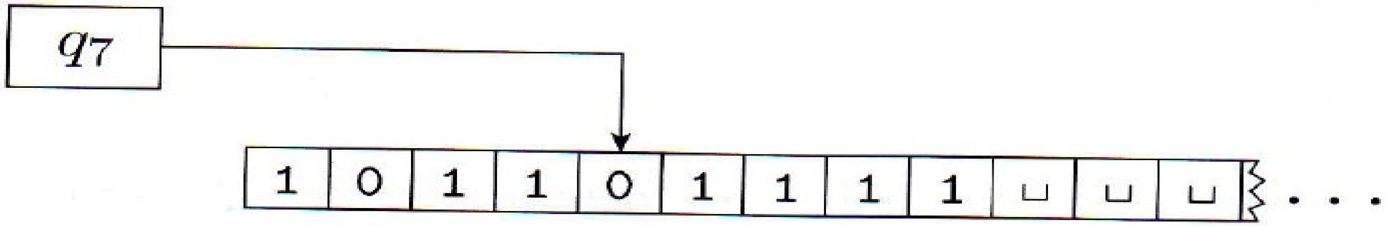
Máquinas de Turing - Funcionamento

- **Configuração** - situação atual da máquina:

Máquinas de Turing - Funcionamento

- **Configuração** - situação atual da máquina:
  - Estado atual
  - Conteúdo da fita
  - Posição da cabeça de fita

- Ex:



Máquinas de Turing - Funcionamento

- **Configuração** - situação atual da máquina:
  - Estado atual
  - Conteúdo da fita
  - Posição da cabeça de fita

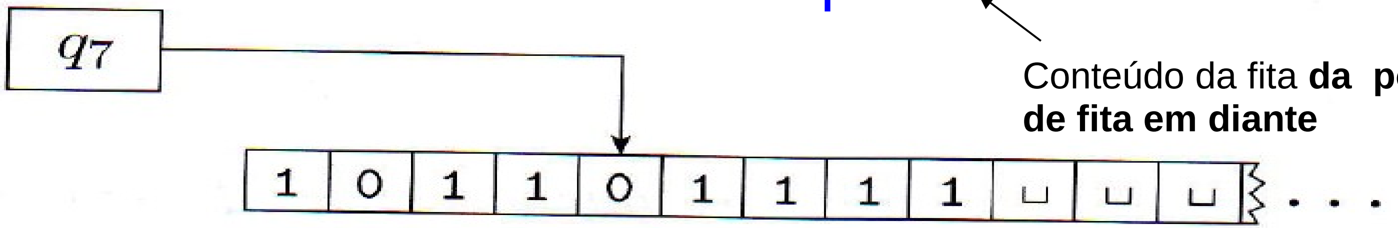
Notação simplificada

Conteúdo da fita **antes** da posição da cabeça de fita

1011 q7 01111

Conteúdo da fita **da posição da cabeça de fita em diante**

- Ex:



Máquinas de Turing - Funcionamento

Dizemos que uma configuração C_1 **origina** uma configuração C_2 se a máquina puder ir de C_1 a C_2 em um **único** passo.

Máquinas de Turing - Funcionamento

$$\boxed{\delta : Q \times \Gamma \longrightarrow Q \times \Gamma \times \{\mathrm{E}, \mathrm{D}\}}$$

Suponha que tenhamos a, b e c em Γ, assim como u e v em Γ^* e os estados q_i e q_j. Nesse caso $ua\,q_i\,bv$ e $u\,q_j\,acv$ são duas configurações. Digamos que

$$ua\,q_i\,bv \quad \text{origina} \quad u\,q_j\,acv$$

se na função de transição

Profa. Ariane Machado Lima

Máquinas de Turing - Funcionamento

$$\boxed{\delta : Q \times \Gamma \longrightarrow Q \times \Gamma \times \{E, D\}}$$

Suponha que tenhamos a, b e c em Γ, assim como u e v em Γ^* e os estados q_i e q_j. Nesse caso $ua\,q_i\,bv$ e $u\,q_j\,acv$ são duas configurações. Digamos que

$ua\,q_i\,bv$ origina $u\,q_j\,acv$

se na função de transição $\delta(q_i, \quad) = (q_j, \qquad)$.

Profa. Ariane Machado Lima

Máquinas de Turing - Funcionamento

$$\boxed{\delta : Q \times \Gamma \longrightarrow Q \times \Gamma \times \{\mathrm{E}, \mathrm{D}\}}$$

Suponha que tenhamos a, b e c em Γ, assim como u e v em Γ^* e os estados q_i e q_j. Nesse caso $ua\,q_i\,bv$ e $u\,q_j\,acv$ são duas configurações. Digamos que

$$ua\,q_i\,bv \quad \text{origina} \quad u\,q_j\,acv$$

se na função de transição $\delta(q_i, b) = (q_j, c, \quad)$.

Máquinas de Turing - Funcionamento

$$\boxed{\delta : Q \times \Gamma \longrightarrow Q \times \Gamma \times \{\mathrm{E}, \mathrm{D}\}}$$

Suponha que tenhamos a, b e c em Γ, assim como u e v em Γ^* e os estados q_i e q_j. Nesse caso $ua\, q_i\, bv$ e $u\, q_j\, acv$ são duas configurações. Digamos que

$$ua\, q_i\, bv \quad \text{origina} \quad u\, q_j\, acv$$

se na função de transição $\delta(q_i, b) = (q_j, c, \mathrm{E})$.

Máquinas de Turing - Funcionamento

$$\boxed{\delta : Q \times \Gamma \longrightarrow Q \times \Gamma \times \{\mathrm{E}, \mathrm{D}\}}$$

Suponha que tenhamos a, b e c em Γ, assim como u e v em Γ^* e os estados q_i e q_j. Nesse caso $ua\, q_i\, bv$ e $u\, q_j\, acv$ são duas configurações. Digamos que

$$ua\, q_i\, bv \quad \text{origina} \quad u\, q_j\, acv$$

se na função de transição $\delta(q_i, b) = (q_j, c, \mathrm{E})$.

$$ua\, q_i\, bv \quad \text{origina} \quad uac\, q_j\, v$$

Máquinas de Turing - Funcionamento

$$\boxed{\delta : Q \times \Gamma \longrightarrow Q \times \Gamma \times \{\mathrm{E}, \mathrm{D}\}}$$

Suponha que tenhamos a, b e c em Γ, assim como u e v em Γ^* e os estados q_i e q_j. Nesse caso $ua\, q_i\, bv$ e $u\, q_j\, acv$ são duas configurações. Digamos que

$$ua\, q_i\, bv \quad \text{origina} \quad u\, q_j\, acv$$

se na função de transição $\delta(q_i, b) = (q_j, c, \mathrm{E})$.

$$ua\, q_i\, bv \quad \text{origina} \quad uac\, q_j\, v$$

se $\delta(q_i, b) = (q_j, c, \mathrm{D})$.

Máquinas de Turing - Funcionamento (cadeia de entrada w)

- Configuração inicial:
- Configuração de aceitação:
- Configuração de rejeição:

Máquinas de Turing - Funcionamento (cadeia de entrada w)

- Configuração inicial: q_0w
- Configuração de aceitação:
- Configuração de rejeição:

Máquinas de Turing - Funcionamento (cadeia de entrada w)

- Configuração inicial: q_0w
- Configuração de aceitação: estado atual = q_{aceita}
- Configuração de rejeição:

Máquinas de Turing - Funcionamento (cadeia de entrada w)

- Configuração inicial: q_0w
- Configuração de aceitação: estado atual = q_{aceita}
- Configuração de rejeição: estado atual = $q_{rejeita}$

Máquinas de Turing - Funcionamento (cadeia de entrada w)

- Configuração inicial: q_0w
- Configuração de aceitação: estado atual = q_{aceita}
- Configuração de rejeição: estado atual = $q_{rejeita}$

Uma máquina de Turing M ***aceita*** a entrada w se uma seqüência de configurações $C_1, C_2, \ldots, C_k$ existe, onde

1. C_1 é a configuração inicial de M sobre a entrada w,
2. cada C_i origina C_{i+1} e
3. C_k é uma configuração de aceitação.

Máquinas de Turing

A coleção de cadeias que M aceita é ***a linguagem de M***, ou ***a linguagem reconhecida por M***, denotada $L(M)$.

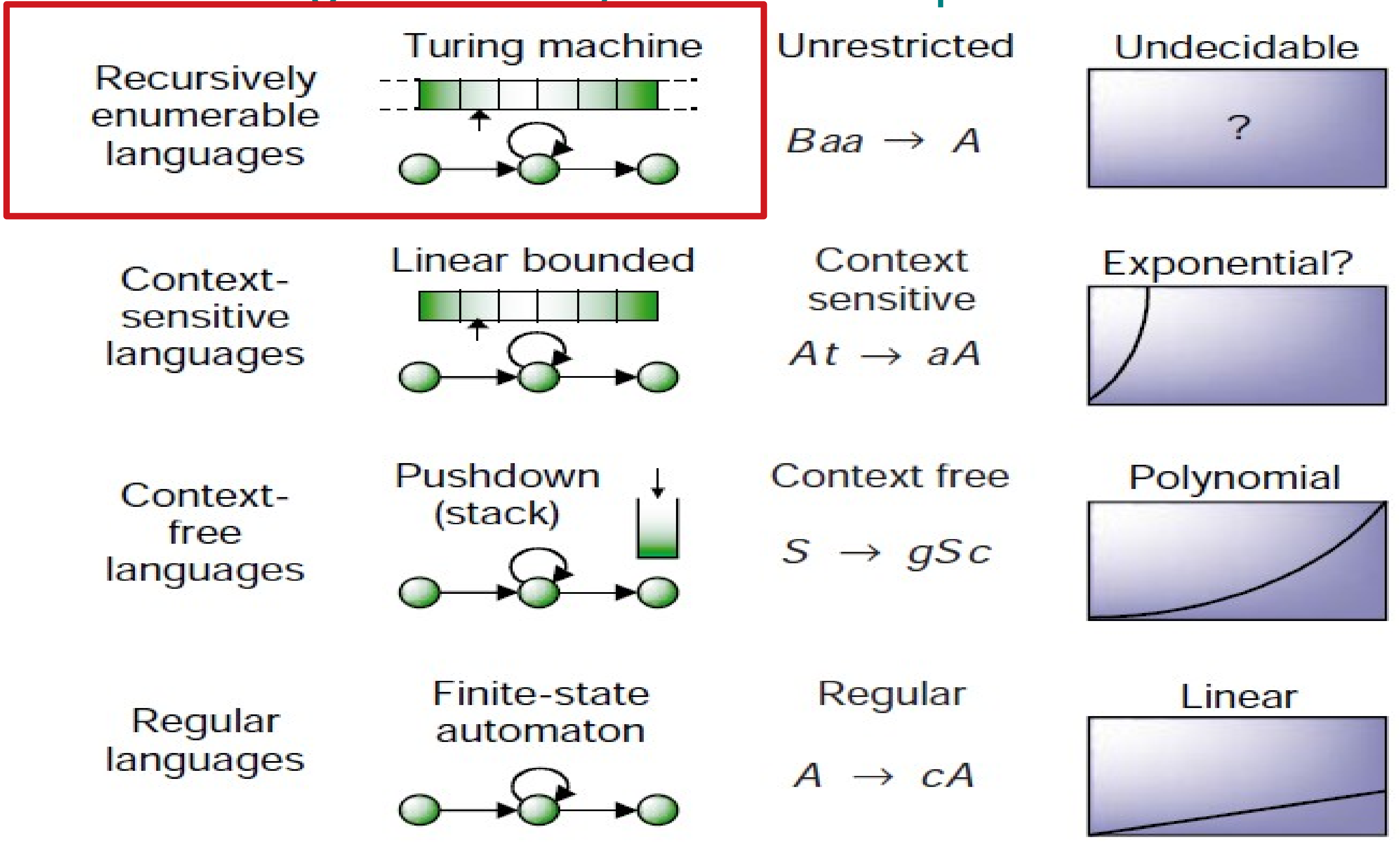
DEFINIÇÃO 3.5

Chame uma linguagem de ***Turing-reconhecível***, se alguma máquina de Turing a reconhece.[1]

1 - Ou linguagem **recursivamente enumerável** ou linguagem **irrestrita**

Profa. Ariane Machado Lima

Linguagens, modelos computacionais (dispositivos, gramáticas) e suas complexidades



Máquinas de Turing (MT) Decisoras

Uma MT é decisora se ela nunca entra em loop (isto é, sempre pára em um estado de aceitação ou de rejeição).

Dizemos que um decisor que reconhece uma linguagem decide essa linguagem.

DEFINIÇÃO 3.6

Chame uma linguagem de ***Turing-decidível*** ou simplesmente ***decidível*** se alguma máquina de Turing a decide.[2]

2 - Ou linguagem **recursiva**

Máquinas de Turing - Exemplos

EXEMPLO 3.7

Aqui descrevemos uma máquina de Turing (MT) M_2 que decide $A = \{0^{2^n} \mid n \geq 0\}$, a linguagem consistindo em todas as cadeias de 0s cujo comprimento é uma potência de 2.

- Ideia: Uma potência de 2, sempre que eu divido por 2, terei outra potência de dois que é um número par ou o número 1

Profa. Ariane Machado Lima

Máquinas de Turing - Exemplos

EXEMPLO 3.7

Aqui descrevemos uma máquina de Turing (MT) M_2 que decide $A = \{0^{2^n} \mid n \geq 0\}$, a linguagem consistindo em todas as cadeias de 0s cujo comprimento é uma potência de 2.

M_2 = "Sobre a cadeia de entrada w:

1. Faça uma varredura da esquerda para a direita na fita, marcando um 0 não, e outro, sim.
2. Se no estágio 1, a fita continha um único 0, *aceite*.
3. Se no estágio 1, a fita continha mais que um único 0 e o número de 0s era ímpar, *rejeite*.
4. Retorne a cabeça para a extremidade esquerda da fita.
5. Vá para o estágio 1."

Exercícios

- Façam os exercícios 3.1 e 3.2 (livro do Sipser) antes da próxima aula!!!

3.1 Este exercício concerne a MT M_2 cuja descrição e diagrama de estados aparecem no Exemplo 3.7. Em cada um dos itens abaixo, dê a seqüência de configurações nas quais M_2 entra quando iniciada sobre a cadeia de entrada indicada:

a. 0.

[R]**b.** 00.

c. 000.

d. 000000.

3.2 Este exercício concerne a MT M_1 cuja descrição e diagrama de estados aparecem no Exemplo 3.9. Em cada um dos itens abaixo, dê a seqüência de configurações nas quais M_1 entra quando iniciada sobre a cadeia de entrada indicada:

[R]**a.** 11.

b. 1#1.

c. 1##1.

d. 10#11.

e. 10#10.