

# Lista de Exercícios

## Autômatos de Pilha (AP)

### Resoluções com Diagramas

Linguagens Formais e Autômatos

2 de dezembro de 2025

## Sumário

<b>1 Exemplo: Linguagem <math>a^n b^n</math></b>	<b>2</b>
<b>2 Exercício 1: Linguagem Vazia</b>	<b>3</b>
<b>3 Exercício 2: Linguagens sobre <math>\Sigma = \{a, b, c\}</math></b>	<b>4</b>
3.1 Exercício 2a: Palavra Vazia . . . . .	4
3.2 Exercício 2b: $i > j$ . . . . .	5
3.3 Exercício 2c: $i < j$ . . . . .	6
3.4 Exercício 2d: $i \neq j$ . . . . .	7
3.5 Exercício 2e: $wcw^R$ . . . . .	8
3.6 Exercício 2f: $k = 2(i + j)$ . . . . .	9
3.7 Exercício 2g: $i = j$ ou $j = k$ . . . . .	10
3.8 Exercício 2h: $a^*wc^n$ . . . . .	11
<b>4 Exercício 3: APD para <math>a^n b^n c^m d^m</math></b>	<b>12</b>
<b>5 Exercício 4: APD para <math>i + j = k + l</math></b>	<b>13</b>
<b>6 Exercício 5: AP para <math>i + j = k + l</math> (com <math>a</math> final)</b>	<b>14</b>
<b>7 Exercício 6: Linguagens Regulares e Livres de Contexto</b>	<b>15</b>
7.1 Exercício 6a: Pelo menos três 1's . . . . .	15
7.2 Exercício 6b: Quantidade ímpar de 0's . . . . .	16
7.3 Exercício 6c: $\Sigma^*$ . . . . .	17
7.4 Exercício 6d: Parênteses Balanceados . . . . .	18
7.5 Exercício 6e: Parênteses e Colchetes Balanceados . . . . .	19
7.6 Exercício 6f: $a^n b^n \cup b^n a^n$ . . . . .	20
7.7 Exercício 6g: Palíndromos de Tamanho Ímpar . . . . .	21
7.8 Exercício 6h: Palíndromos . . . . .	22
<b>8 Resumo dos Exercícios</b>	<b>23</b>

## 1 Exemplo: Linguagem $a^n b^n$

### Enunciado

Construa um AP que reconheça a linguagem  $L = \{a^n b^n \mid n \geq 0\}$

### Estratégia

Empilha um símbolo para cada  $a$  lido, depois desempilha um para cada  $b$ . Aceita se a pilha ficar vazia (apenas  $Z$ ) ao final.

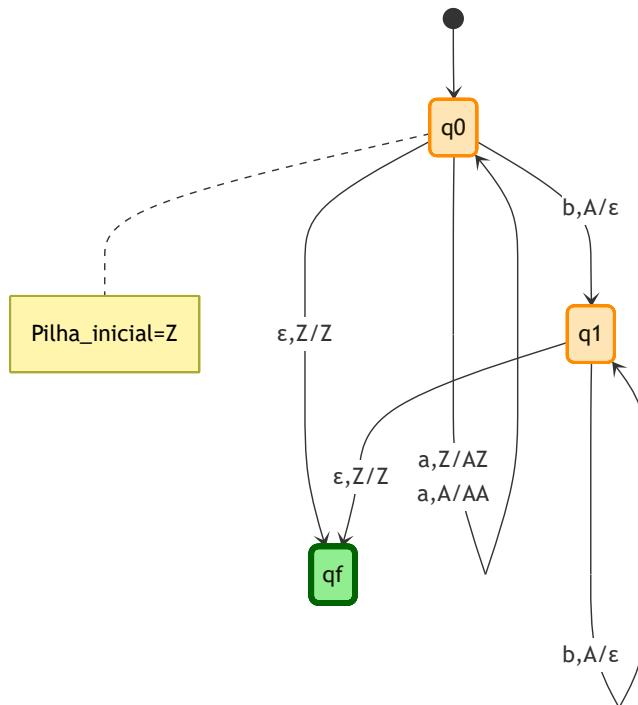


Figura 1: AP Exemplo:  $L = \{a^n b^n\}$

## 2 Exercício 1: Linguagem Vazia

### Enunciado

Construa um AP que reconheça a seguinte linguagem:  $L = \emptyset$

### Estratégia

Um AP que reconhece a linguagem vazia não possui nenhum estado final acessível. Qualquer entrada será rejeitada.

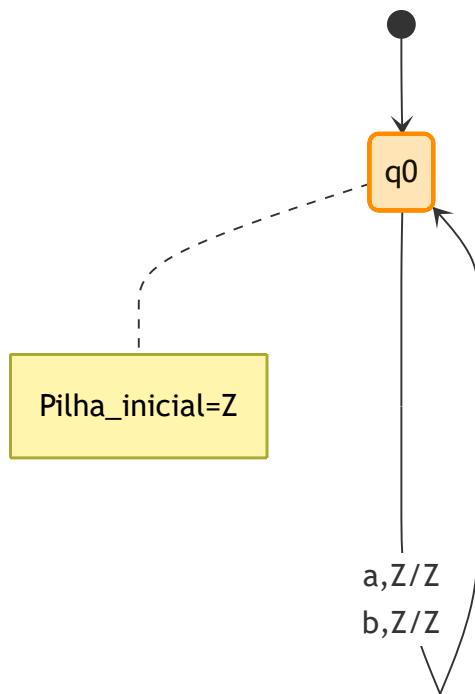


Figura 2: AP 1:  $L = \emptyset$

### 3 Exercício 2: Linguagens sobre $\Sigma = \{a, b, c\}$

#### 3.1 Exercício 2a: Palavra Vazia

##### Enunciado

$L = \{w \mid w = \lambda\}$  (apenas a palavra vazia)

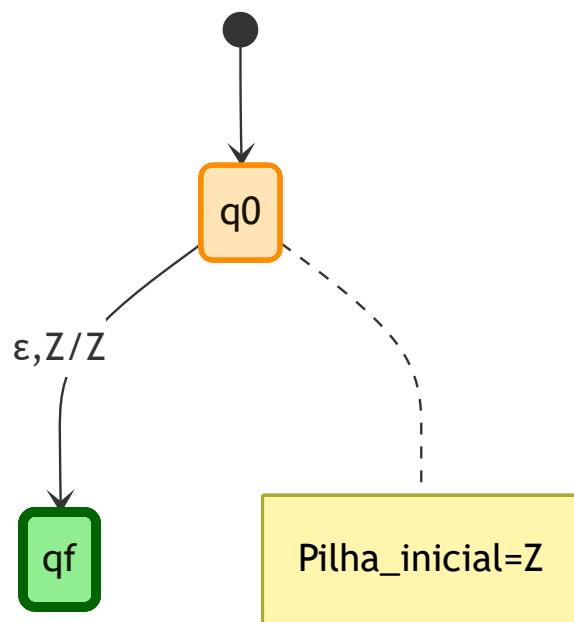


Figura 3: AP 2a:  $L = \{\lambda\}$

### 3.2 Exercício 2b: $i > j$

#### Enunciado

$$L = \{a^i b^j \mid i > j\}$$

#### Estratégia

Empilha  $A$  para cada  $a$ , desempilha para cada  $b$ . Aceita se ainda houver  $A$ 's na pilha ao final (mais  $a$ 's que  $b$ 's).

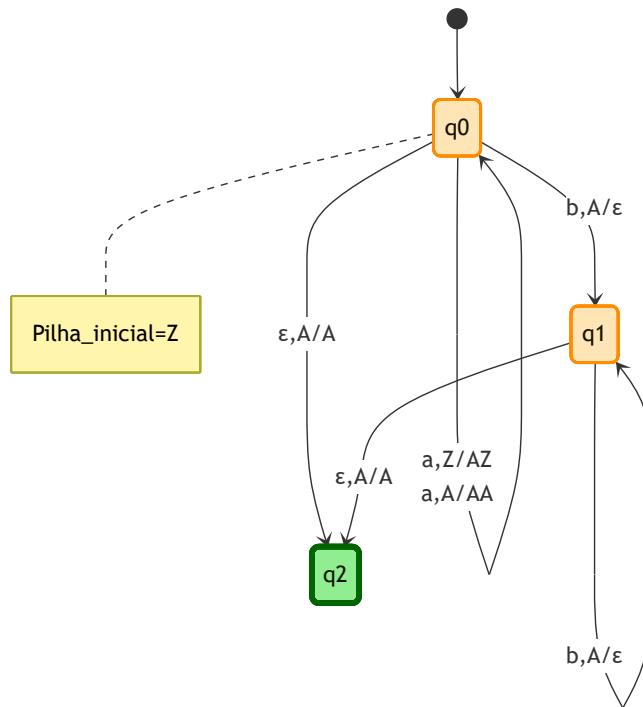


Figura 4: AP 2b:  $L = \{a^i b^j \mid i > j\}$

### 3.3 Exercício 2c: $i < j$

#### Enunciado

$$L = \{a^i b^j \mid i < j\}$$

#### Estratégia

Empilha  $A$  para cada  $a$ , desempilha para cada  $b$ . Após consumir todos os  $a$ 's, continua aceitando  $b$ 's extras (empilhando  $B$ 's). Aceita se houver  $B$ 's na pilha ao final.

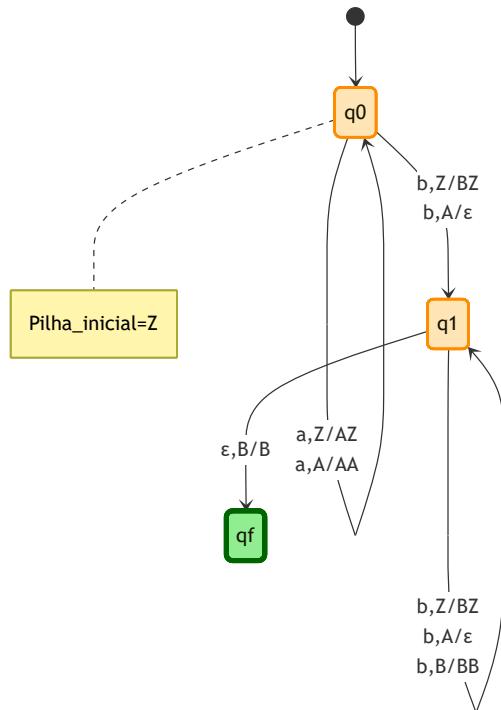


Figura 5: AP 2c:  $L = \{a^i b^j \mid i < j\}$

### 3.4 Exercício 2d: $i \neq j$

#### Enunciado

$$L = \{a^i b^j \mid i \neq j\}$$

#### Estratégia

União de  $i > j$  e  $i < j$ . O AP não-determinístico escolhe qual ramo seguir.

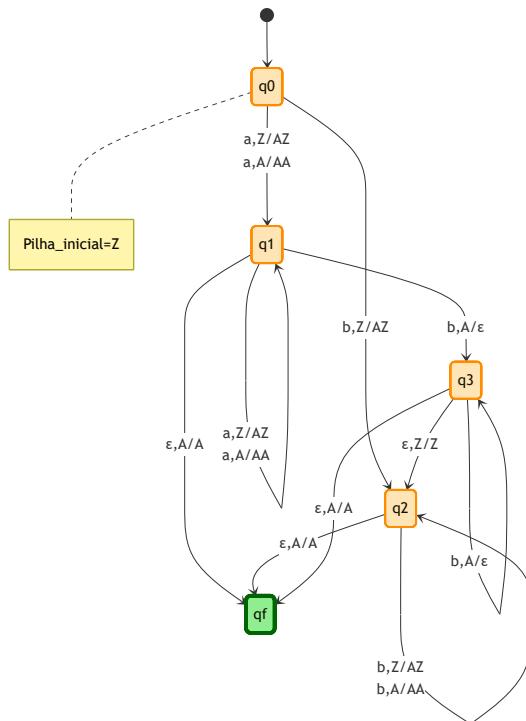


Figura 6: AP 2d:  $L = \{a^i b^j \mid i \neq j\}$

### 3.5 Exercício 2e: $wcw^R$

#### Enunciado

$$L = \{wcw^R \mid w \in \{a, b\}^*\}$$

#### Estratégia

Empilha todos os símbolos antes do  $c$ . Após ler  $c$ , desempilha comparando com os símbolos restantes. Aceita se tudo casar (palíndromo com marcador central).

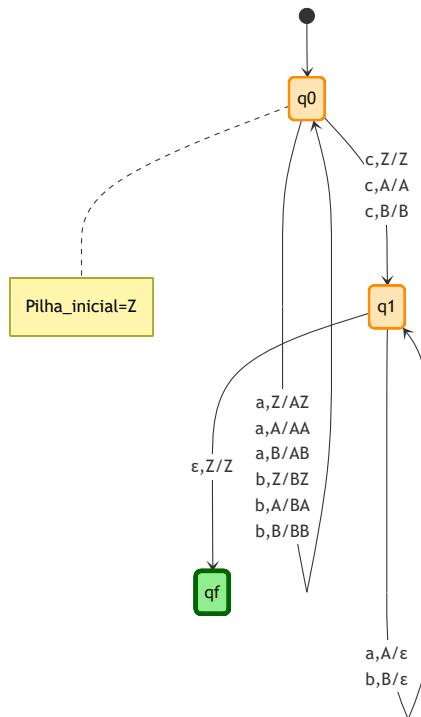


Figura 7: AP 2e:  $L = \{wcw^R\}$

### 3.6 Exercício 2f: $k = 2(i + j)$

#### Enunciado

$$L = \{a^i b^j c^k \mid i, j, k > 0 \text{ e } k = 2(i + j)\}$$

#### Estratégia

Para cada  $a$  ou  $b$ , empilha 2 símbolos (representando que cada um contribui 2 para o total de  $c$ 's necessários). Depois desempilha um para cada  $c$ .

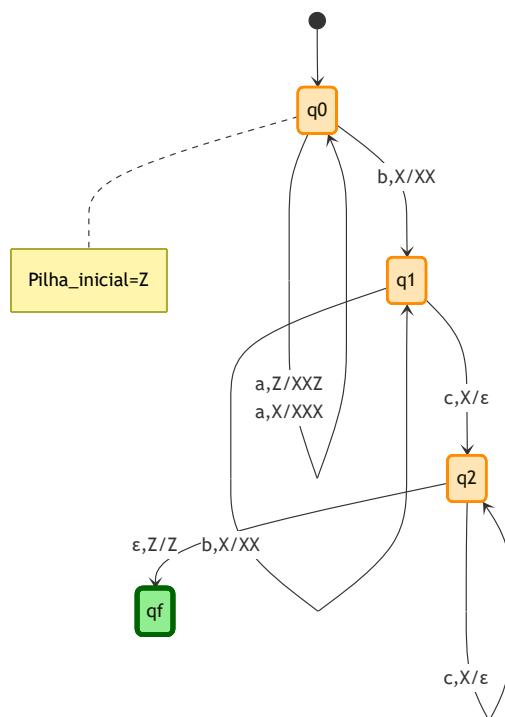


Figura 8: AP 2f:  $L = \{a^i b^j c^k \mid k = 2(i + j)\}$

### 3.7 Exercício 2g: $i = j$ ou $j = k$

#### Enunciado

$$L = \{a^i b^j c^k \mid i, j, k > 0 \text{ e } (i = j \text{ ou } j = k)\}$$

#### Estratégia

AP não-determinístico com duas ramificações:

- Ramo 1 ( $i = j$ ): Empilha para cada  $a$ , desempilha para cada  $b$ , ignora  $c$ 's
- Ramo 2 ( $j = k$ ): Ignora  $a$ 's, empilha para cada  $b$ , desempilha para cada  $c$

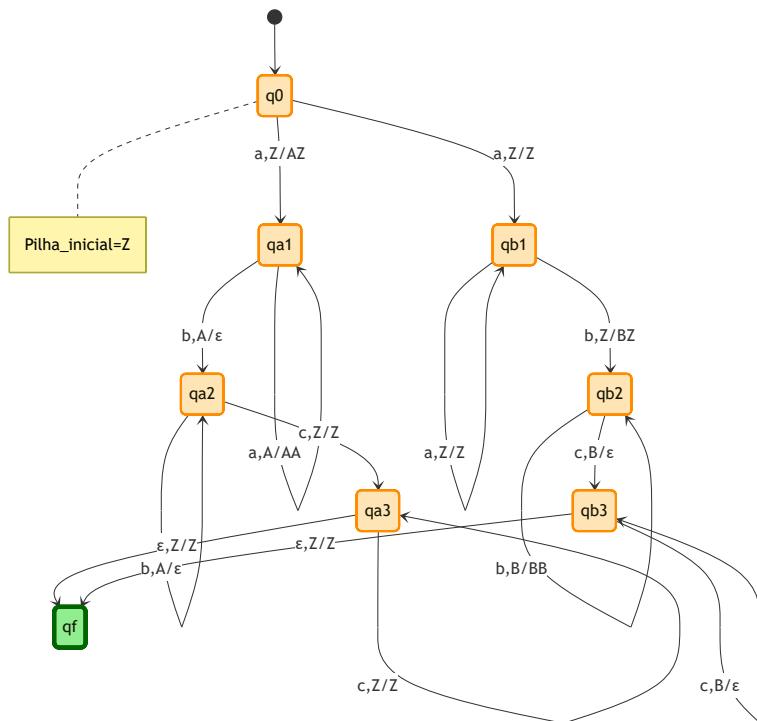


Figura 9: AP 2g:  $L = \{a^i b^j c^k \mid i = j \vee j = k\}$

### 3.8 Exercício 2h: $a^*wc^n$

#### Enunciado

$L = \{a^*wc^n \mid w \in \{a,b\}^* \text{ e } n = |w|_a\}$   
 (Número de  $c$ 's igual ao número de  $a$ 's em  $w$ )

#### Estratégia

O prefixo  $a^*$  é ignorado (não conta). Para a parte  $w$ , empilha um símbolo para cada  $a$  encontrado (ignora  $b$ 's). Depois desempilha um para cada  $c$ .

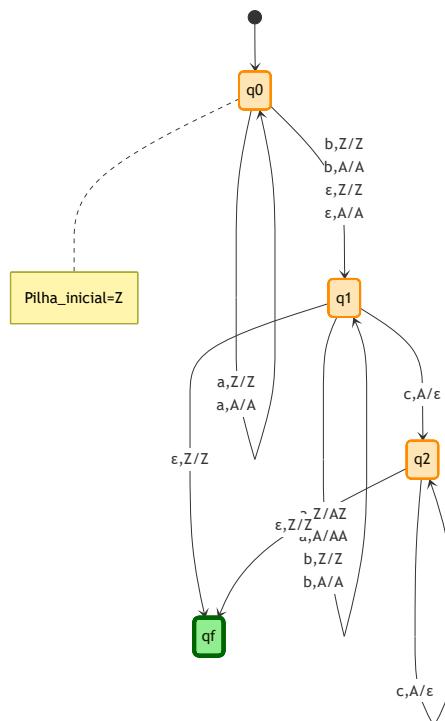


Figura 10: AP 2h:  $L = \{a^*wc^n \mid n = |w|_a\}$

## 4 Exercício 3: APD para $a^n b^n c^m d^m$

### Enunciado

Construa um APD que reconheça a seguinte linguagem:

$$L = \{a^n b^n c^m d^m \mid n, m \in \mathbb{N}\}$$

### Estratégia

O autômato processa em duas fases independentes:

1. Empilha para cada  $a$ , desempilha para cada  $b$  (garante  $n$   $a$ 's e  $n$   $b$ 's)
2. Empilha para cada  $c$ , desempilha para cada  $d$  (garante  $m$   $c$ 's e  $m$   $d$ 's)

Como as fases são independentes, é determinístico.

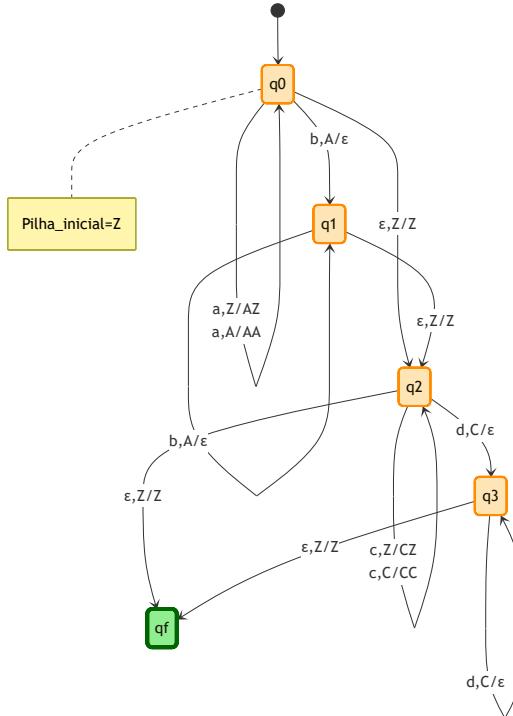


Figura 11: APD 3:  $L = \{a^n b^n c^m d^m\}$

## 5 Exercício 4: APD para $i + j = k + l$

### Enunciado

Construa um APD que reconheça a seguinte linguagem:

$$L = \{a^i b^j c^k d^l \mid i + j = k + l \text{ e } i, j, k, l \in \mathbb{N}\}$$

### Estratégia

Empilha um símbolo para cada  $a$  e cada  $b$  (acumulando  $i + j$ ). Depois desempilha um para cada  $c$  e cada  $d$  (consumindo  $k + l$ ). Aceita se pilha voltar ao estado inicial.

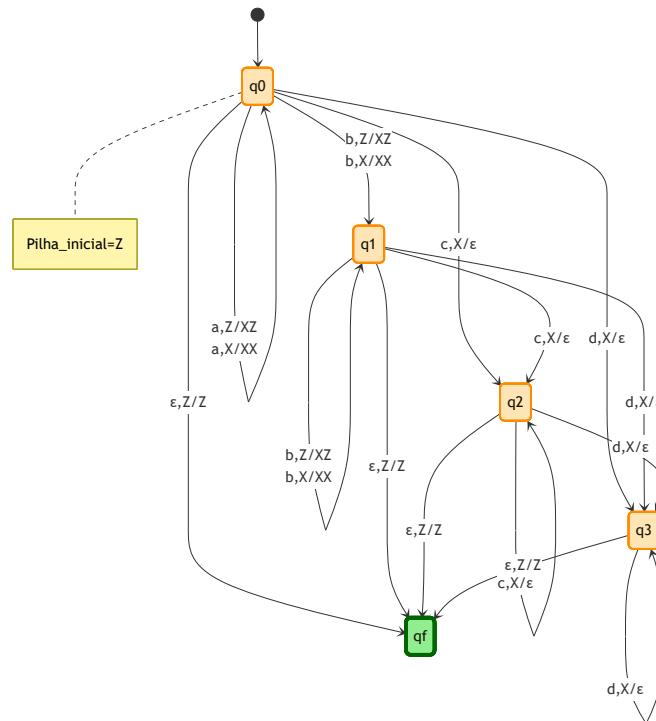


Figura 12: APD 4:  $L = \{a^i b^j c^k d^l \mid i + j = k + l\}$

## 6 Exercício 5: AP para $i + j = k + l$ (com $a$ final)

### Enunciado

Construa um AP que reconheça a seguinte linguagem:

$$L = \{a^i b^j c^k a^l \mid i + j = k + l \text{ e } i, j, k, l \in \mathbb{N}\}$$

### Estratégia

Similar ao exercício 4, mas o último símbolo é  $a$  em vez de  $d$ . Empilha para  $a$ 's e  $b$ 's iniciais, desempilha para  $c$ 's e  $a$ 's finais.

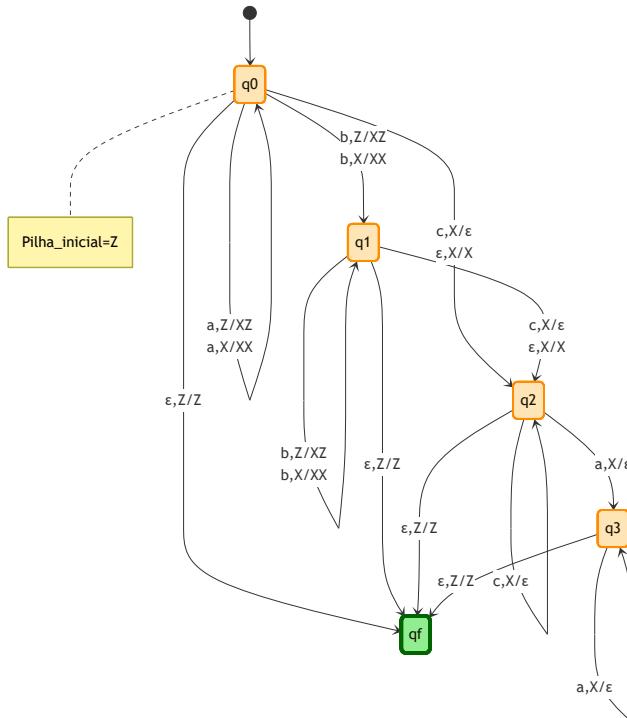


Figura 13: AP 5:  $L = \{a^i b^j c^k a^l \mid i + j = k + l\}$

## 7 Exercício 6: Linguagens Regulares e Livres de Contexto

### 7.1 Exercício 6a: Pelo menos três 1's

#### Enunciado

$L = \{w \in \{0,1\}^* \mid w \text{ possui pelo menos três 1's}\}$

#### Observação

Esta é uma linguagem regular! A pilha não é utilizada. O AP funciona como um AFD, contando os 1's com os estados.

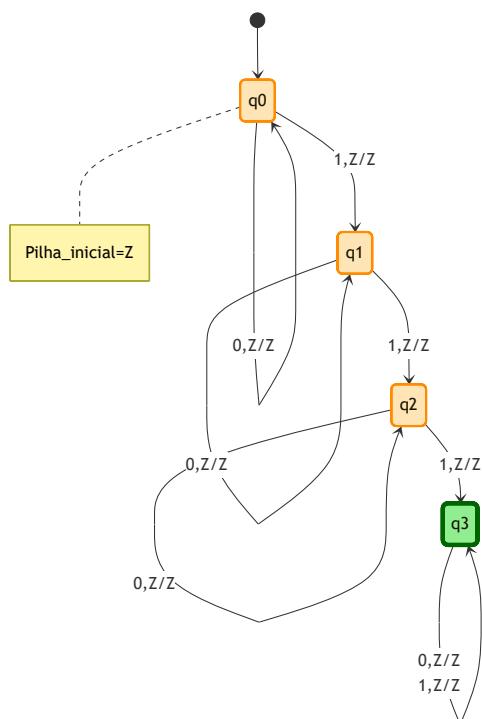


Figura 14: AP 6a: Pelo menos três 1's

## 7.2 Exercício 6b: Quantidade ímpar de 0's

### Enunciado

$L = \{w \in \{0, 1\}^* \mid w \text{ possui quantidade ímpar de 0's}\}$

### Observação

Linguagem regular. O AP alterna entre dois estados a cada 0 lido, sem usar a pilha.

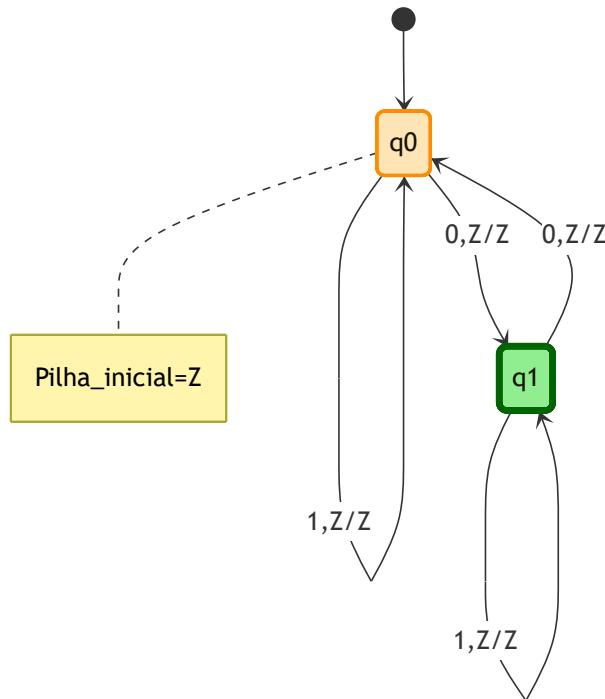


Figura 15: AP 6b: Quantidade ímpar de 0's

### 7.3 Exercício 6c: $\Sigma^*$

#### Enunciado

$L = \{w \in \{0,1\}^* \mid w \text{ pode ser descrito como a ER } (0^*1^*)^*\}$

#### Observação

$(0^*1^*)^* = \Sigma^*$  (todas as cadeias sobre  $\{0,1\}$ ). O AP aceita qualquer entrada sem usar a pilha.

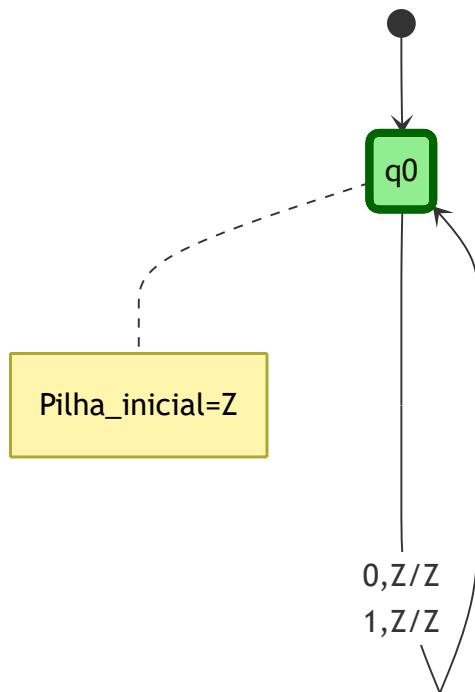


Figura 16: AP 6c:  $\Sigma^*$

## 7.4 Exercício 6d: Parênteses Balanceados

### Enunciado

$L = \{w \mid w \text{ é formado por parênteses balanceados}\}$

Exemplo: ( ( ) ) ( ) ( ( ) )

### Estratégia

Empilha um símbolo para cada (, desempilha para cada ). Aceita se a pilha estiver vazia ao final e nunca ficar “negativa”.

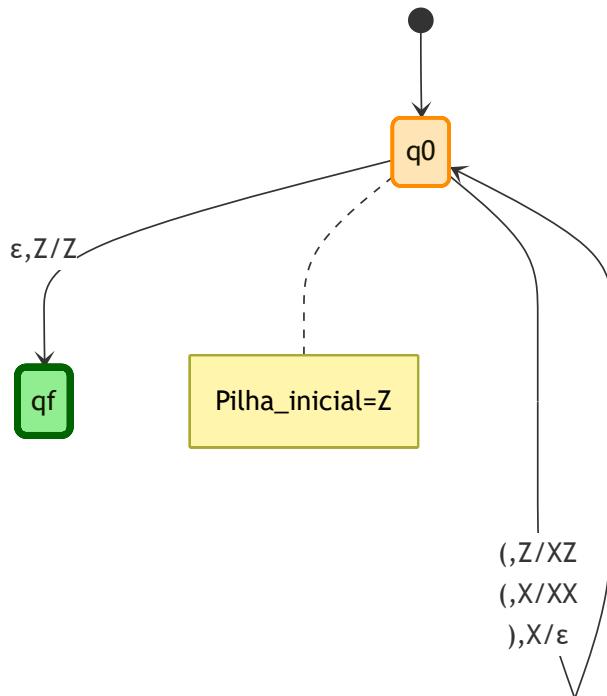


Figura 17: AP 6d: Parênteses Balanceados

## 7.5 Exercício 6e: Parênteses e Colchetes Balanceados

### Enunciado

$L = \{w \mid w \text{ é formado por parênteses e colchetes balanceados}\}$

Exemplo: [ ( [ ( ) ] ) ( ) ] ( ( ) [ ( ) ] )

### Estratégia

Empilha símbolos diferentes para ( e [. Ao ler ) ou ], verifica se o topo da pilha corresponde ao tipo correto de abertura.

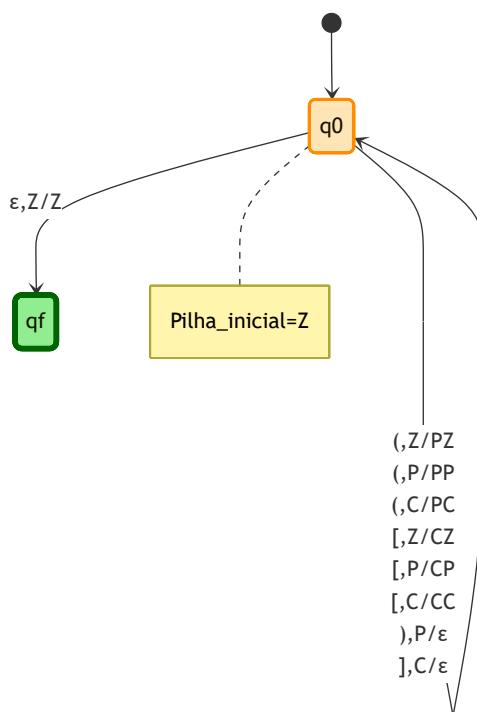


Figura 18: AP 6e: Parênteses e Colchetes Balanceados

## 7.6 Exercício 6f: $a^n b^n \cup b^n a^n$

### Enunciado

$L = \{w \mid w \text{ é formado por } n \text{ } a's \text{ seguidos de } n \text{ } b's, \text{ ou } n \text{ } b's \text{ seguidos de } n \text{ } a's\}$

### Estratégia

AP não-determinístico que escolhe entre duas ramificações:

- Ramo 1: Empilha  $a$ 's, desempilha com  $b$ 's
- Ramo 2: Empilha  $b$ 's, desempilha com  $a$ 's

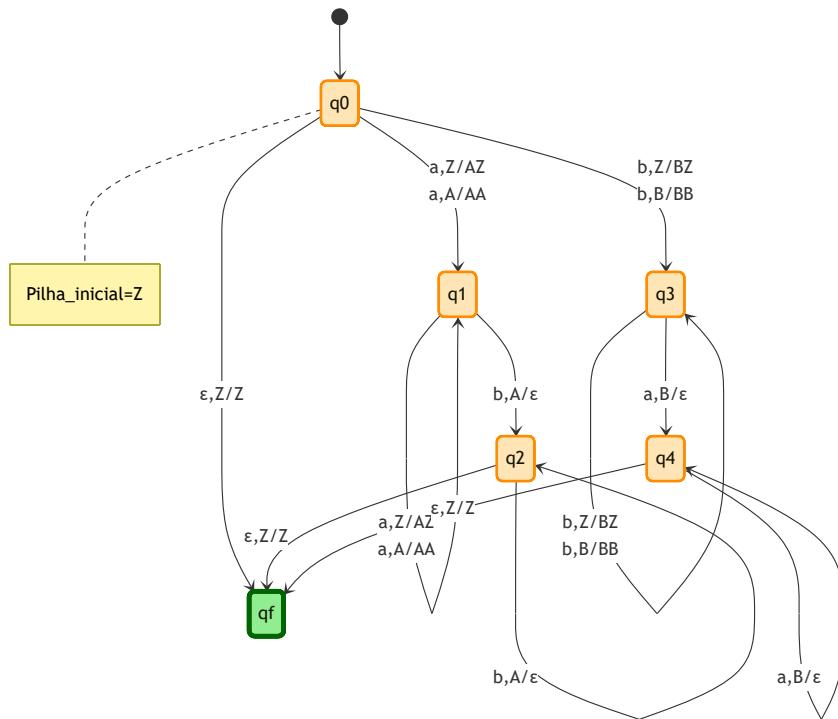


Figura 19: AP 6f:  $a^n b^n \cup b^n a^n$

## 7.7 Exercício 6g: Palíndromos de Tamanho Ímpar

### Enunciado

$$L = \{w \in \{0, 1\}^* \mid w = w^R \text{ e } |w| \text{ é ímpar}\}$$

### Estratégia

Empilha símbolos até não-deterministicamente decidir que está no meio (lendo o símbolo central). Depois desempilha verificando se os símbolos casam.

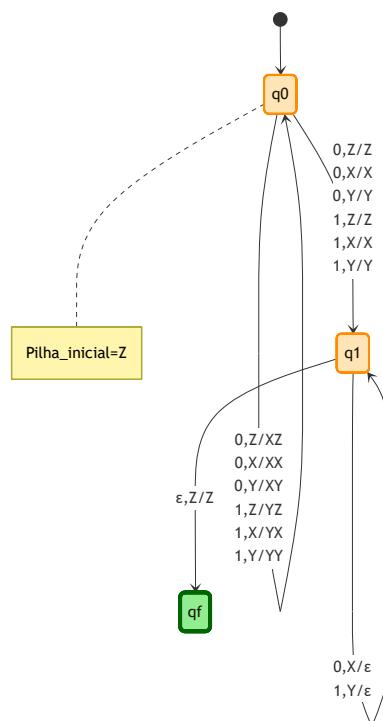


Figura 20: AP 6g: Palíndromos de Tamanho Ímpar

## 7.8 Exercício 6h: Palíndromos

### Enunciado

$L = \{w \in \{0,1\}^* \mid w = w^R\}$  (todos os palíndromos)

### Estratégia

Similar ao 6g, mas também aceita palíndromos de tamanho par. O AP não-deterministicamente escolhe quando está no meio:

- Para tamanho par: transição  $\epsilon$  para fase de desempilhamento
- Para tamanho ímpar: lê símbolo central sem alterar pilha

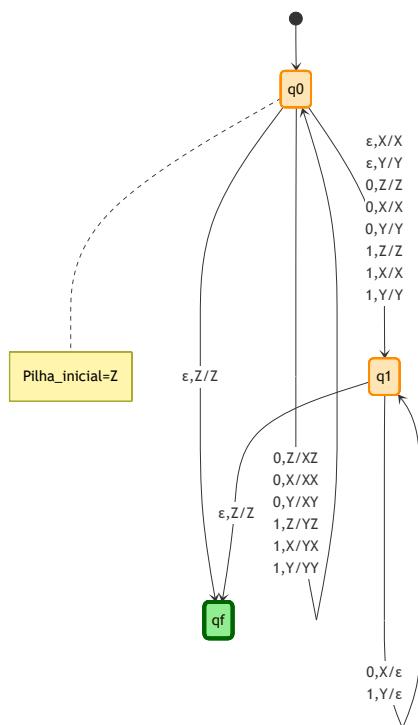


Figura 21: AP 6h: Palíndromos

## 8 Resumo dos Exercícios

Ex.	Linguagem	Alfabeto	Tipo
Ex	$a^n b^n$	$\{a, b\}$	LLC
1	$L = \emptyset$	$\{a, b, c\}$	-
2a	$L = \{\lambda\}$	$\{a, b, c\}$	Regular
2b	$a^i b^j, i > j$	$\{a, b, c\}$	LLC
2c	$a^i b^j, i < j$	$\{a, b, c\}$	LLC
2d	$a^i b^j, i \neq j$	$\{a, b, c\}$	LLC
2e	$wcw^R$	$\{a, b, c\}$	LLC
2f	$a^i b^j c^k, k = 2(i + j)$	$\{a, b, c\}$	LLC
2g	$a^i b^j c^k, i = j \vee j = k$	$\{a, b, c\}$	LLC
2h	$a^* w c^n, n =  w _a$	$\{a, b, c\}$	LLC
3	$a^n b^n c^m d^m$	$\{a, b, c, d\}$	LLC (APD)
4	$a^i b^j c^k d^l, i + j = k + l$	$\{a, b, c, d\}$	LLC (APD)
5	$a^i b^j c^k a^l, i + j = k + l$	$\{a, b, c\}$	LLC
6a	Pelo menos 3 uns	$\{0, 1\}$	Regular
6b	Quantidade ímpar de 0's	$\{0, 1\}$	Regular
6c	$\Sigma^*$	$\{0, 1\}$	Regular
6d	Parênteses平衡ados	$\{(, )\}$	LLC (APD)
6e	Parênteses e colchetes平衡ados	$\{(, ), [ , ]\}$	LLC (APD)
6f	$a^n b^n \cup b^n a^n$	$\{a, b\}$	LLC
6g	Palíndromos ímpares	$\{0, 1\}$	LLC
6h	Palíndromos	$\{0, 1\}$	LLC

### Legenda:

- **LLC**: Linguagem Livre de Contexto (requer pilha)
- **APD**: Autômato de Pilha Determinístico
- **Regular**: Linguagem regular (pilha não é necessária)

## Observações

- Todos os APs foram validados com testes aleatórios
- Os diagramas foram gerados automaticamente usando Mermaid
- Os arquivos JSON de definição estão em `inputAP/`
- Os diagramas PDF estão em `diagramasAP/`
- Notação de transições:  $\sigma, X/\alpha$  significa “lendo  $\sigma$ , com  $X$  no topo, substitui por  $\alpha$ ”
- $\varepsilon$  representa transição vazia (sem consumir entrada)