E-fólio B | Folha de resolução para E-fólio

UNIDADE CURRICULAR: Linguagens e Computação

CÓDIGO: 21078

DOCENTES: Jorge Morais e Rúdi Gualter (tutor)

A preencher pelo estudante

NOME:

N.º DE ESTUDANTE:

CURSO:

DATA DE ENTREGA:

TRABALHO / RESOLUÇÃO:

1. Considere um saco com bolas azuis e vermelhas, de onde se tiram as mesmas uma a uma, e onde se pensa que o número de bolas vermelhas não é inferior ao de bolas azuis. Finda a tiragem, pretende-se confirmar este último facto, construindo uma gramática independente de contexto (CFG) para o efeito.

No exercício 5 da AF4 fizemos a resolução para o caso em que o número de 0's e 1's são iguais. A resolução é quase igual com uma pequena alteração. Começando por transcrever a abordagem 1 da resolução referida:

"Vamos analisar as sequências pares que pertencem à linguagem.

A palavra vazia pertence à linguagem, vou ter de ter sempre, sendo S o estado inicial, que conseguir derivar ε a partir de S (posso ter S-->ε, ou S-->P e P-->ε, etc.).

01 e 10 são as palavras de tamanho 2 que pertencem à linguagem. São facilmente geráveis usando produções S-->0S1|1S0. Contrariamente às capicuas, os símbolos à esquerda e à direita são opostos.

Continuando, agora para o tamanho 4:

0011, 0101, 0110, 1001, 1010, 1100

Se pegarmos nas instruções S-->0S1|1S0|ε, em 4 dos casos, conseguimos gerar as sequências: 0011, 0101, 1010 e 1100. O problema é precisamente com as capicuas. Vamos ver as sequências de tamanho 6:

000111, 001011, 001101, 001110, 010011, 010101, 010110, 011001,011010, 011100, 100011, 100101, 100110, 101001, 101010, 101100, 110001, 110010, 110100, 111000

Vamos pegar nos que não são reconhecidos pelas produções S-->0S1|1S0|ε:

01-10

10-01

0011-01

0011-10

01-0011

0101-10

01-1010

01-1100

10-0011

10-0101

1010-01

10-1100

1100-01

1100-10

Fiz aqui uma separação para vermos que cada uma das partes pode ser reconhecida pelas referidas produções, mas não o todo.

Se pensarmos numa sequência grande aleatória, digamos, com tamanho 20, como:

01011001100101100110

Vamos tentar separar como fizemos anteriormente:

0101-100110-0101-100110

(note-se que pode haver mais do que uma hipótese de separação, por exemplo 0101 é válida, mas 01 em particular também é)

Embora possa não ser imediato, eu vou sempre conseguir, para sequências em que o número de 0´s e 1´s é igual, ter 1 ou mais subsequências geradas por aquelas produções e que garantem que o número de 0's e 1's é igual. Para isso, se tiver S-->SS conjugado com S-->ε, eu posso ter qualquer número de subsequências.

Chegamos assim à CFG final:

S-->SS|0S1|1S0|ε

Como vimos, esta CFG será ambígua, basta ver o caso 0101:

S ==> 0S1 ==> 0101

ou

S ==> SS ==> 0S1S => 01S ==> 010S1 => 0101

ou ainda

S ==> 0S1 ==> 0SS1 ==> 01S0S1 ==> 010S1 ==> 0101"

Terminada a transcrição, agora a diferença é que teremos um número maior ou igual de bolas vermelhas comparativamente às azuis. Trocando os 0's e 1's por a e v, temos, para o caso da igualdade:

S → SS | aSv | vSa | ε

Existindo mais ocorrências de v, bastará acrescentar S → v.

S → SS | aSv | vSa | v | ε

2. Teste a gramática para a tiragem: vermelha, vermelha, vermelha, azul, azul, vermelha, azul, azul, vermelha, azul, vermelha.

Pode haver várias soluções, dado a gramática ser ambígua. Uma hipótese seria:

S => (3 x S→SS)

SSSS => (S→v)

vSSS => (2 x S→vSa)

vvvSaaSS => (S→ ε)

vvvaaSS => (S→vSa)

vvvaavSaS => (S→ ε)

vvvaavaS => (S→aSv)

vvvaavaaSv => (S→vSa)

vvvaavaavSav => (S→ ε)

vvvaavaavav

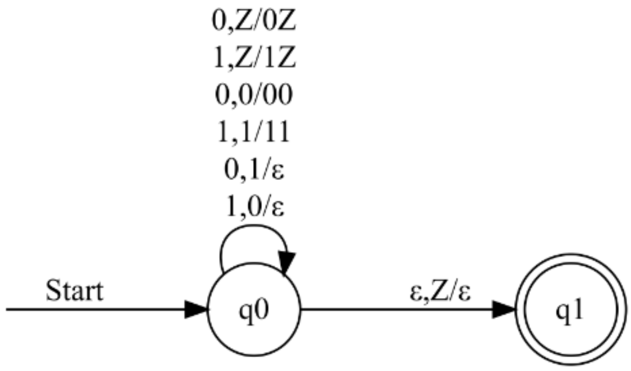
3. Entretanto, diz uma lenda que uma fada boa resolveu intervir, fazendo com que o número de bolas azuis e vermelhas fosse igual. Mas mal virou costas, a fada má resolveu introduzir 3 bolas brancas no saco. Esse saco ficou guardado num baú durante muitos anos, até ser descoberto por um informático que, sabendo da lenda, resolveu criar um autómato de pilha (PDA) para verificar se aquele poderia ser, efetivamente, o saco da lenda. Descreva esse PDA.

A resolução do exercício 5 da AF4 também dá uma pista. No caso, temos o mesmo número de bolas vermelhas e azuis, apenas não temos as brancas. Começando por transcrever a resolução do referido exercício:

“A forma mais simples é eu pensar em empilhar o símbolo quando a pilha tem no topo o símbolo inicial ou quando o elemento no topo é igual, e desempilhar quando o símbolo é diferente. Assim, a pilha será composto por n elementos de um dos símbolos se existirem, até ao momento, n ocorrências a mais desse símbolo relativamente ao outro. Por exempo, 001110, contando o topo da pilha como o elemento mais à esquerda, teremos sucessivamente:

Z, 0Z, 00Z, 0Z, Z, 1Z, Z e, finalmente, a pilha vazia.

Assim, o PDA será este:

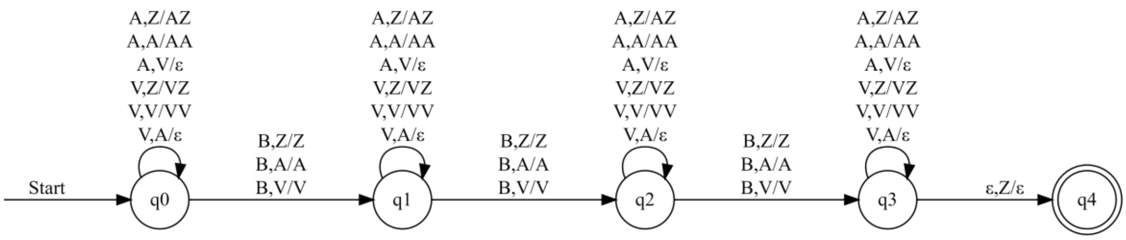


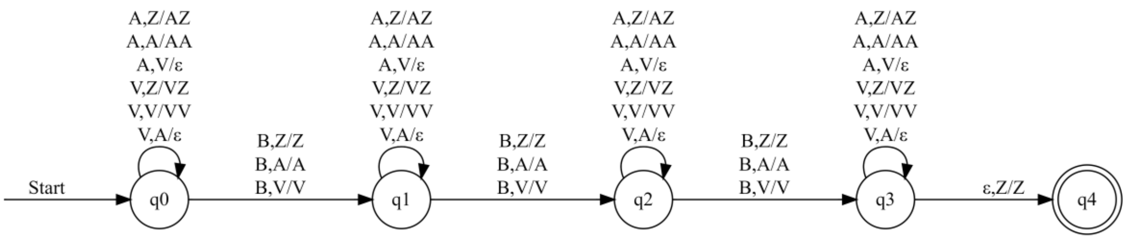
Note-se que aqui reconheço simultaneamente por pilha vazia e por estado final.

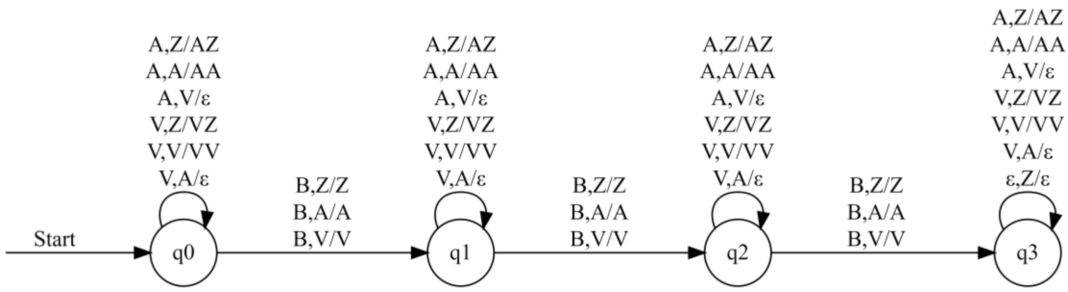
Se quisesse apenas por estado final, poderia deixar como está ou então fazer $\delta(q0, \varepsilon, Z) = (q1, Z)$ - não mexendo na pilha, deixava o Z no topo.

Se quisesse apenas por pilha vazia, não haveria necessidade de existir o estado final, pelo que poderia ficar δ(q0, ε, Z) = (q0, ε) - retirando o Z, a pilha fica vazia e a sequência é aceite, não havendo necessidade de mudar de estado.

Neste caso concreto, temos 3 bolas brancas misturadas com um número igual de azuis e vermelhas. Podemos manter o mesmo princípio, mas sempre que sair uma bola branca mudamos de estado sem mexer na pilha. Assim, quando estivermos nos estados q0, q1, q2 e q3, teremos tirado 0,1, 2 e 3 bolas brancas, respetivamente. Só a partir do momento em que estamos em q3 é que poderemos terminar, caso o número de bolas azuis e vermelhas ser igual, ficando no topo da pilha o símbolo inicial Z. Vejamos a solução para as 3 situações de reconhecimento (pilha vazia + estados finais, estados finais e pilha vazia, respetivamente):



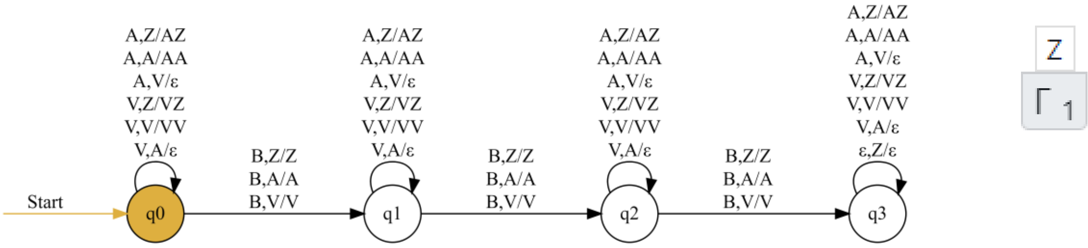




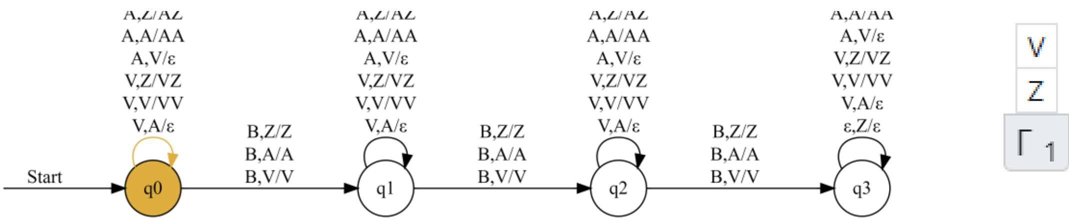
4. Usando a ferramenta UAbALL, verifique que a sequência vermelha, branca, vermelha, vermelha, azul, branca, azul, vermelha, azul, branca, azul, vermelha, azul. Mostre a simulação passo, incluindo o diagrama e a pilha.

Vamos pegar no exemplo de reconhecimento por pilha vazia:

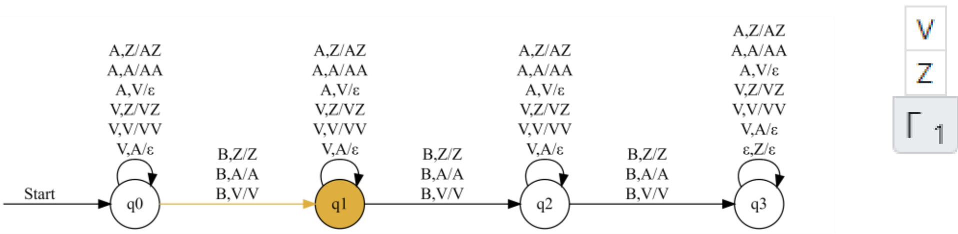
Início: VBVVABAVABAVA



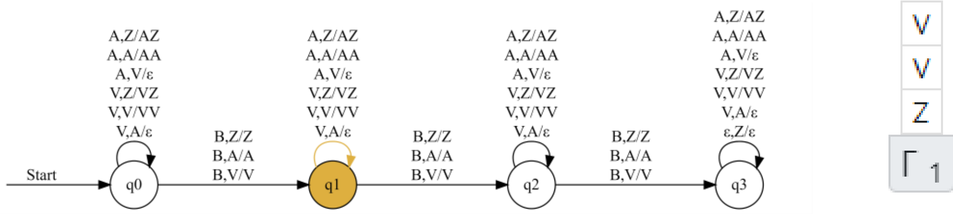
Início: **V**BVVABAVABAVA



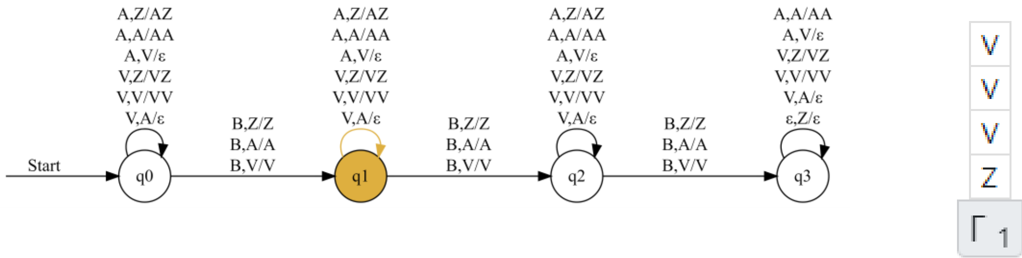
Início: V**B**VVABAVABAVA



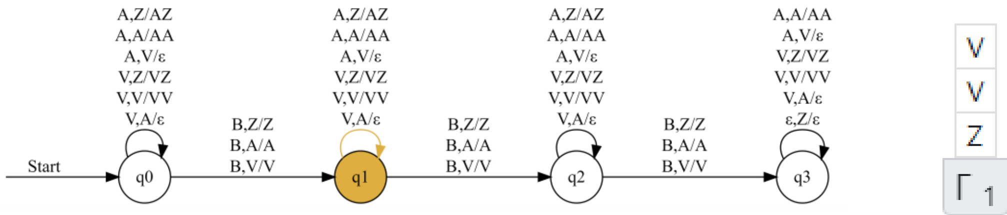
Início: VB**V**VABAVABAVA



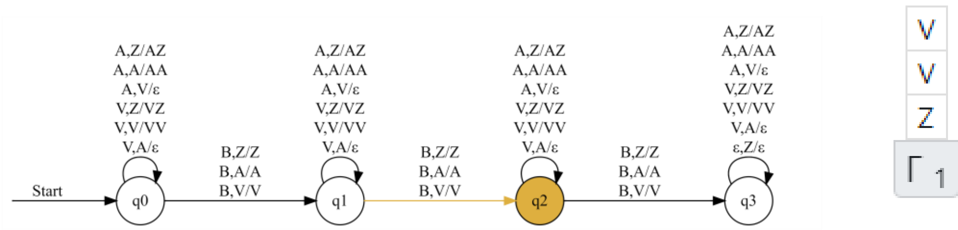
Início: VBV**V**ABAVABAVA



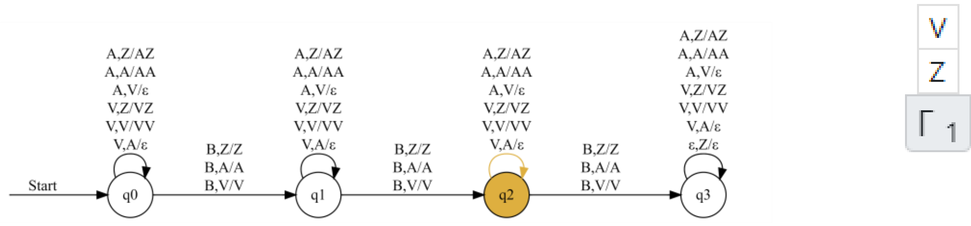
Início: VBVV**A**BAVABAVA



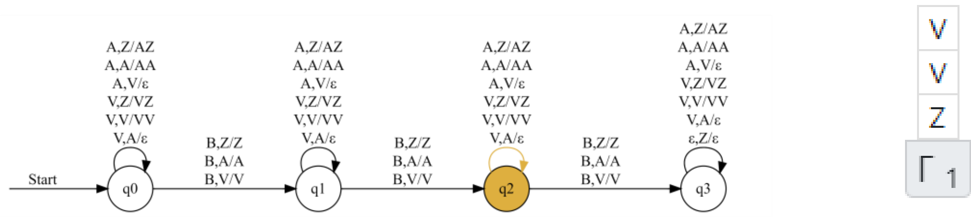
Início: VBVVA**B**AVABAVA



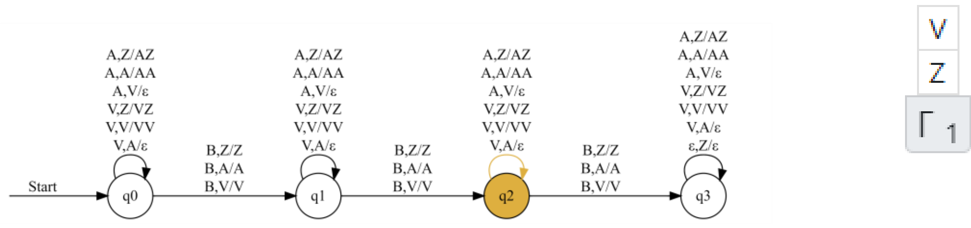
Início: VBVVAB**A**VABAVA



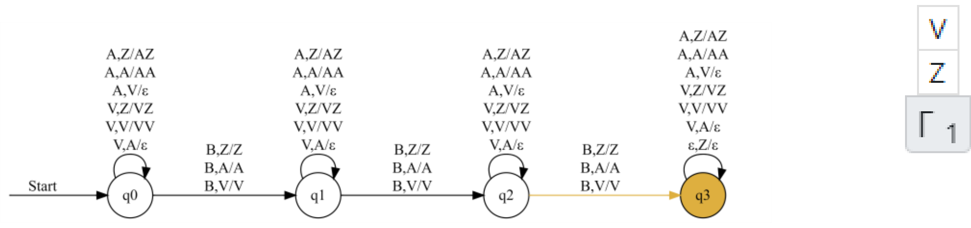
Início: VBVVABA**V**ABAVA



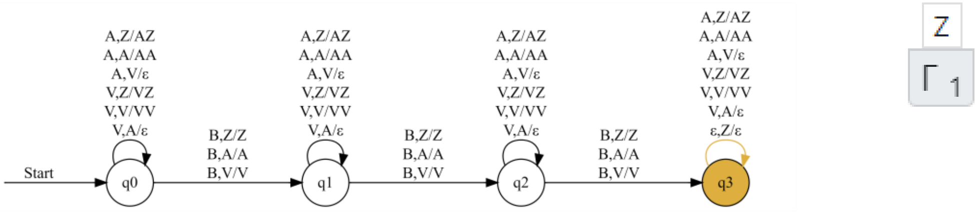
Início: VBVVABAV**A**BAVA



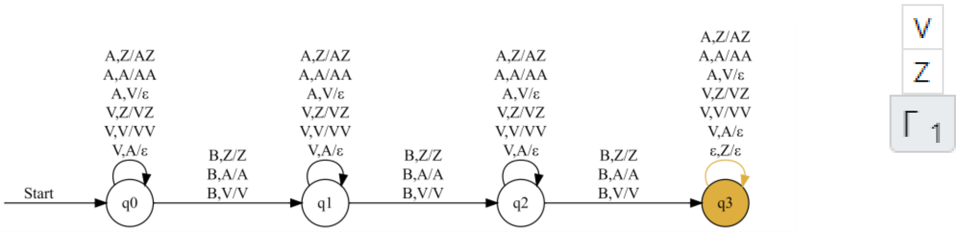
Início: VBVVABAVA**B**AVA



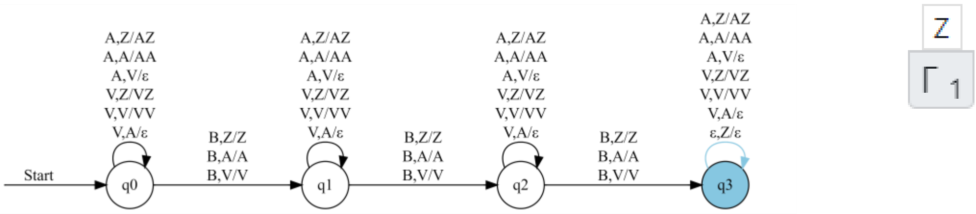
Início: VBVVABAVAB**A**VA



Início: VBVVABAVABA**V**A



Início: VBVVABAVABAV**A**



Início: VBVVABAVABAVA