E-fólio B | Instruções para a realização do E-fólio

UNIDADE CURRICULAR: Linguagens e Computação

CÓDIGO: 21078

DOCENTE: Jorge Morais

NOME: Paulo Jorge Martins Nicolau

N.º DE ESTUDANTE: 1800465

CURSO: Licenciatura em Engenharia Informática

DATA DE ENTREGA: 4 de Janeiro de 2021

TRABALHO / RESOLUÇÃO:

Considere o alfabeto ∑ = {0,1}.

1. **Escreva uma gramática independente de contexto que reconheça todas as sequências da forma $0^{2n}1^{k}0^{n-k}$, onde n e k são números inteiros positivos e k ≤ n.**

Como n e k são números inteiros positivos e k terá que ser menor ou igual a n, então o menor valor possível para eles é 1. Desta forma a menor palavra que esta gramática poderá reconhecer será dado pela seguinte expressão $0^{2\times1}1^{1}0^{1-1}=0^{2}1^{1}0^{0}$ e que corresponde à palavra 001.

Segue-se uma lista de palavras que a gramática deverá aceitar:

n	**k**	**Palavra**	**Expressão**
1	1	001	$0^{2\times1}1^{1}0^{1-1}=0^{2}1^{1}0^{0}$
2	1	000010	$0^{2\times2}1^{1}0^{2-1}=0^{4}1^{1}0^{1}$
2	2	000011	$0^{2\times2}1^{2}0^{2-2}=0^{4}1^{2}0^{0}$
3	1	000000100	$0^{2\times3}1^{1}0^{3-1}=0^{6}1^{1}0^{2}$
7	5	0000000000000011111 00	$0^{2\times7}1^{5}0^{7-5}=0^{14}1^{5}0^{2}$

Assim, a seguinte gramática independente de contexto, é uma proposta que reconhece todas as sequências da forma $0^{2n}1^{k}0^{n-k}$, e é definida por :

G = (V,T,P,S) :

V = {S,A}

T = {0,1}

P = {S → 00S0 | 00A1 | 001, A → 00A1 | 001 }

Pela definição da gramática verifica-se que S é a variável inicial da gramática da qual será gerado todas as sequências. Existem 5 produções no total, de modo a remover produções ε, que só seriam permitidas na variável inicial S, no entanto a produção S → 001 e a inexistência de uma produção vazia de S garante que a menor palavra que a gramática reconhece será 001, o que faz com que a palavra vazia não seja permitida nesta gramática.

A existência na gramática das produções S → 00S0 e S → 00A1 garante que a quantidade de 0 após o último 1 da palavra será sempre a quantidade obtida do cálculo n-k.

De modo a verificar que a gramática proposta realmente reconhece as palavras anteriores, segue-se os seguintes exemplos de execução, com a lista de palavras apresentadas anteriormente.

001

S → **001**	Produção:	(S→001)

000010

S → **00S0**	Produção:	(S→00S0)
→ 00**001**0		(S→001)

000011

S → **00A1**	Produção:	(S→00A1)
→ 00**001**1		(A→001)

000000100

S → **00S0**	Produção:	(S→00S0)
→ 00**00S0**0		(S→00S0)
→ 0000**001**00		(S→001)

000000000000001111100

S → **00S0**	Produção:	(S→00S0)
→ 00**00S0**0		(S→00S0)
→ 0000**00A1**00		(S→00A1)
→ 000000**00A1**100		(A→00A1)
→ 00000000**00A1**1100		(A→00A1)
→ 0000000000**00A1**11100		(A→00A1)
→ 000000000000**001**111100		(A→001)

Como é possível verificar nos exemplos anteriores, a gramática desenvolvida permite gerar as sequências da linguagem solicitada.

2. **Construa um autómato de pilha que reconheça a mesma linguagem (sem recorrer à gramática da questão anterior).**

O autómato de pilha com a seguinte definição, permite reconhecer a mesma linguagem:

$P=(Q,\Sigma,\tau,\delta,q0,Z,F)$:

$Q=\{q0,q1,q2,q3,q4\}$

$\Sigma=\{0,1\}$

$\tau=\{Z,0\}$

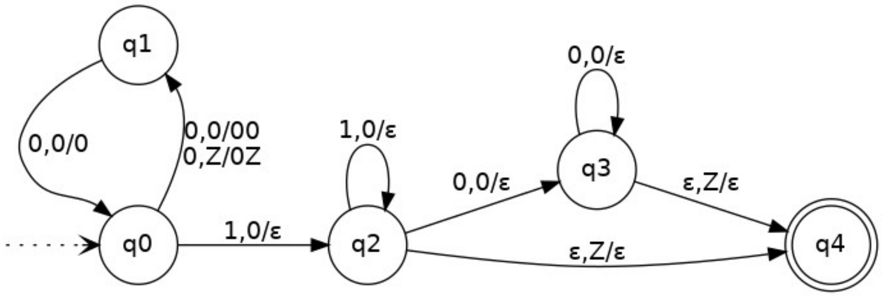
$\delta=Conjunto\ Finito\ de\ Transições$

$F=\{q4\}$

Em que o conjunto finito de transições são as seguintes:

$$\begin{aligned}
&\delta(q0,0,Z)=\{(q1,0Z)\}\\
&\delta(q0,0,0)=\{(q1,00)\}\\
&\delta(q0,1,0)=\{(q2,\epsilon)\}\\
&\delta(q1,0,0)=\{(q0,0)\}\\
&\delta(q2,1,0)=\{(q2,\epsilon)\}\\
&\delta(q2,\epsilon,Z)=\{(q4,\epsilon)\}\\
&\delta(q2,0,0)=\{(q3,\epsilon)\}\\
&\delta(q3,0,0)=\{(q3,\epsilon)\}\\
&\delta(q3,\epsilon,Z)=\{(q4,\epsilon)\}
\end{aligned}$$

E o diagrama do autómato de pilha é o que se segue:



O autómato obtido é um autómato de pilha com reconhecimento por pilha vazia, pois ao ler o final da palavra, irá limpar a pilha, caso esta seja reconhecida. Como também chega a um estado final, o autómato de pilha também pode reconhecer a palavra por estado final.

Este autómato inicia no estado q0 e irá aceitar 0s, alternando entre os estados q0 e q1, onde nas transições de q0 para q1 vai adicionar 0 na pilha, e na transição de q1 para q0 vai manter a pilha igual.

Assim que começar a receber 1s o autómato passará para o estado q2, onde enquanto receber 1s vai verificar a existência de 0s na pilha e remover um elemento. Caso após os 1s começar a receber 0s, irá passar para o estado q3 verificando e removendo 0s da pilha enquanto receber 0. Visto que inicialmente a pilha se encontra com o símbolo Z, se ao chegar ao final da palavra a pilha se encontrar vazia, ou seja sem o símbolo Z, a palavra é reconhecida pelo autómato.

De forma a validar o autómato desenvolvido segue-se dois testes efetuados com duas palavras, uma aceite pela linguagem e outra não aceite.

Palavra inválida: 0010

Pilha Inicial			Pilha Final		
Z	**Input:** 0 **Estado Inicial:** q0 **Transição:** (0,Z/0Z)		0 Z	**Estado Final:**	q1
0 Z	**Input:** 0 **Estado Inicial:** q1 **Transição:** (0,0/0)		0 Z	**Estado Final:**	q0
0 Z	**Input:** 1 **Estado Inicial:** q0 **Transição:** (1,0/ε)		 Z	**Estado Final:**	q2
Z	**Input:** 0 **Estado Inicial:** q2 **Transição:** Não Existe		Z	**Estado Final:**	Fica em q2 devido ao erro e não reconhece a palavra

Como não existe transição válida não é possível continuar a execução, não sendo aceite a palavra.

Palavra válida: 000010

Pilha Inicial	Input	Estado Inicial	Transição	Pilha Final	Estado Final
Z	0	q0	(0,Z/0Z)	0 Z	q1
0 Z	0	q1	(0,0/0)	0 Z	q0
0 Z	0	q0	(0,0/00)	0 0 Z	q1
0 0 Z	0	q1	(0,0/0)	0 0 Z	q0
0 0 Z	1	q0	(1,0/ε)	0 Z	q2
0 Z	0	q2	(0,0/ε)	Z	q3
Z	ε	q3	(ε,0/ε)		q4

Como se verifica a pilha final vazia a palavra foi aceite.

3. **Usando uma das respostas anteriores, mostre que 000000000011000 pertence à respetiva linguagem.**

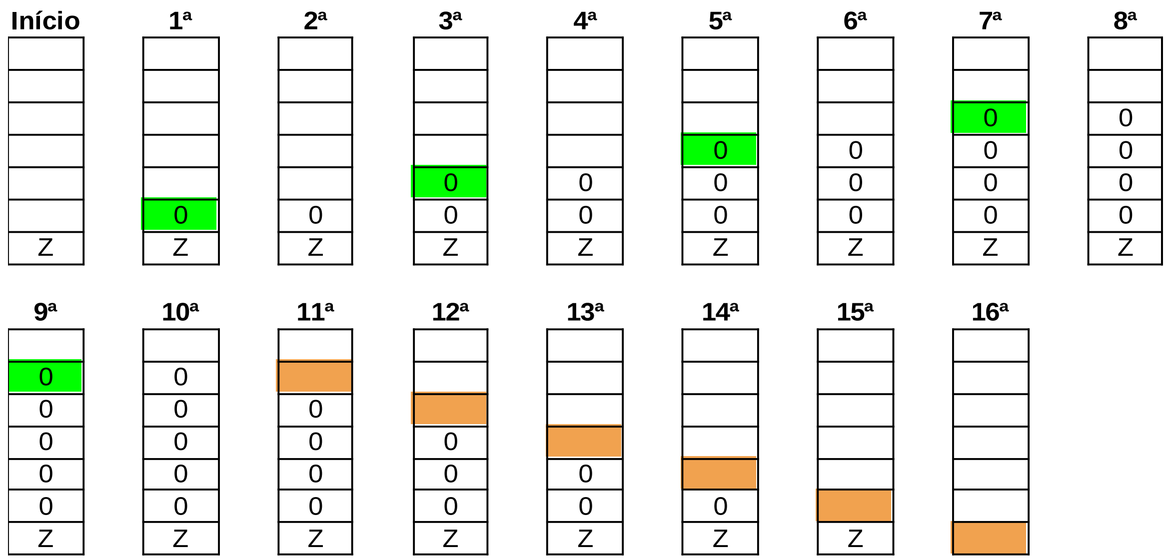
Optei por verificar que a palavra 000000000011000 pertence à linguagem anteriormente indicada através do autómato de pilha desenvolvido na questão 2.

Em seguida encontra-se uma tabela com as operações efetuadas durante a execução do autómato.

Execução	Input	Estado Inicial	Transição	Estado Final
1ª	0	q0	(0,Z/0Z)	q1
2ª	0	q1	(0,0/0)	q0
3ª	0	q0	(0,0/00)	q1
4ª	0	q1	(0,0/0)	q0
5ª	0	q0	(0,0/00)	q1
6ª	0	q1	(0,0/0)	q0
7ª	0	q0	(0,0/00)	q1
8ª	0	q1	(0,0/0)	q0
9ª	0	q0	(0,0/00)	q1
10ª	0	q1	(0,0/0)	q0
11ª	1	q0	(1,0/ε)	q2
12ª	1	q2	(1,0/ε)	q2
13ª	0	q2	(0,0/ε)	q3
14ª	0	q3	(0,0/ε)	q3
15ª	0	q3	(0,0/ε)	q3
16ª	ε	q3	(ε,Z/ε)	q4

Em seguida é apresentado um esquema visual do estado da pilha em cada execução referida na tabela, identificada pelo número da execução:

Estado da Pilha ao longo da Execução do Autómato



Como é possível verificar, após a leitura da palavra, a pilha ficou vazia, assim, esta é reconhecida pelo autómato e pertence à linguagem que contém todas as sequências da forma $0^{2n}1^{k}0^{n-k}$.

Na realização deste trabalho foi utilizado o livro da bibliografia da UC[1] e também recursos disponíveis em outras plataformas[2]

1 **Livro da Bibliografia da UC:** Hopcroft, Motwani & Ullman. *Introduction to Automata Theory, Languages and Computation, 3rd edition*. Addison-Wesley.

2 **Plataformas Utilizadas:**

- Canal Youtube – Aulas de Computação – Playlist [33 a 39]
- Website Tutorialspoint – Automata Theory – Pushdown Automata Acceptance