Lista de Exercícios 2

Teoria da Computação — Mestrado em Computação Aplicada
Prof. Jefferson O. Andrade

Aluno: **Bruno Kobi Valadares de Amorim**

2022/2

1. Exercício

Seja CountGs um problema computacional que recebe I, uma string ASCII, como entrada. A solução (única) é o número de vezes que "G" ocorre em I, escrito em notação decimal.

1.a.

O CountGs tem alguma instância positiva? Se sim, dê um exemplo.

R: Sim, CountGs não tem retorno de "no"em nenhuma combinação de entrada ASCII, então é uma instância positiva. conforme o Livro:"For problems defined on ASCII inputs, an instance is negative if its solution set is the singleton "no". Otherwise the instance is positive."

```
def CountGs(I):
    total = I.count("G")
    potencia = 0
    while total >= 10:
        potencia = potencia + 1
        total = total / 10
        print(potencia)
        continue
    return str(total)+'X10^'+str(potencia)
```

1.b.

O CountGs tem alguma instância negativa? Se sim, dê um exemplo.

R: Não

1.c.

Sugira uma definição de CountGsDecision, um problema de decisão que intuitivamente parece exigir o mesmo processo computacional que o problema da função CountGs.

```
def CountGsDecision(I):
    if I.count("G") > 0:
        return 'yes'
    else:
        return 'no'
```

2. Exercício

Seja L uma linguagem decidível

2.a.

Prove que $\bar{L}$ também é decidível.

R: $\bar{L}$ = {(B,W) | B é um AFD que aceita W de entrada}
Se tanto L quanto $\bar{L}$ são Turing-reconhecíveis, fazemos M1 ser o reconhecedor para L e M2 o reconhecedor para $\bar{L}$.
W = "aa"
M1 = ((q1,q2).(a,b),((q1,a,q2),(q1,b,q2),(q2,a,q2),(q2,b,q2)),q1,(q2))
Testando W na máquina de turing M1 = Aceito
M2 = ((q1,q2).(not a, not b),((q1,not a,q2),(q1,not b,q2),(q2,a,q2),(q2,not b,q2)),q1,(q2))
Testando W na máquina de turing M2 = Rejeitado
Provando que $\bar{L}$ também é decidível.

2.b.

Prove que a união e a interseção de linguagens decidível também são decidível

R: Para provar esse teorema construimos um novo AFC C apartir de A e B , tal que C aceite somente aquelas cadeias são aceitas ou por A ou por B, não por ambos Se A e B reconhecem a mesma linguagem C não aceitará nada.
MtA = ((q1,q2).(a),((q1,a,q2),(q2,a,q2)),q1,(q2))
MtB = ((q1,q2).(b),((q1,b,q2),(q2,b,q2)),q1,(q2))

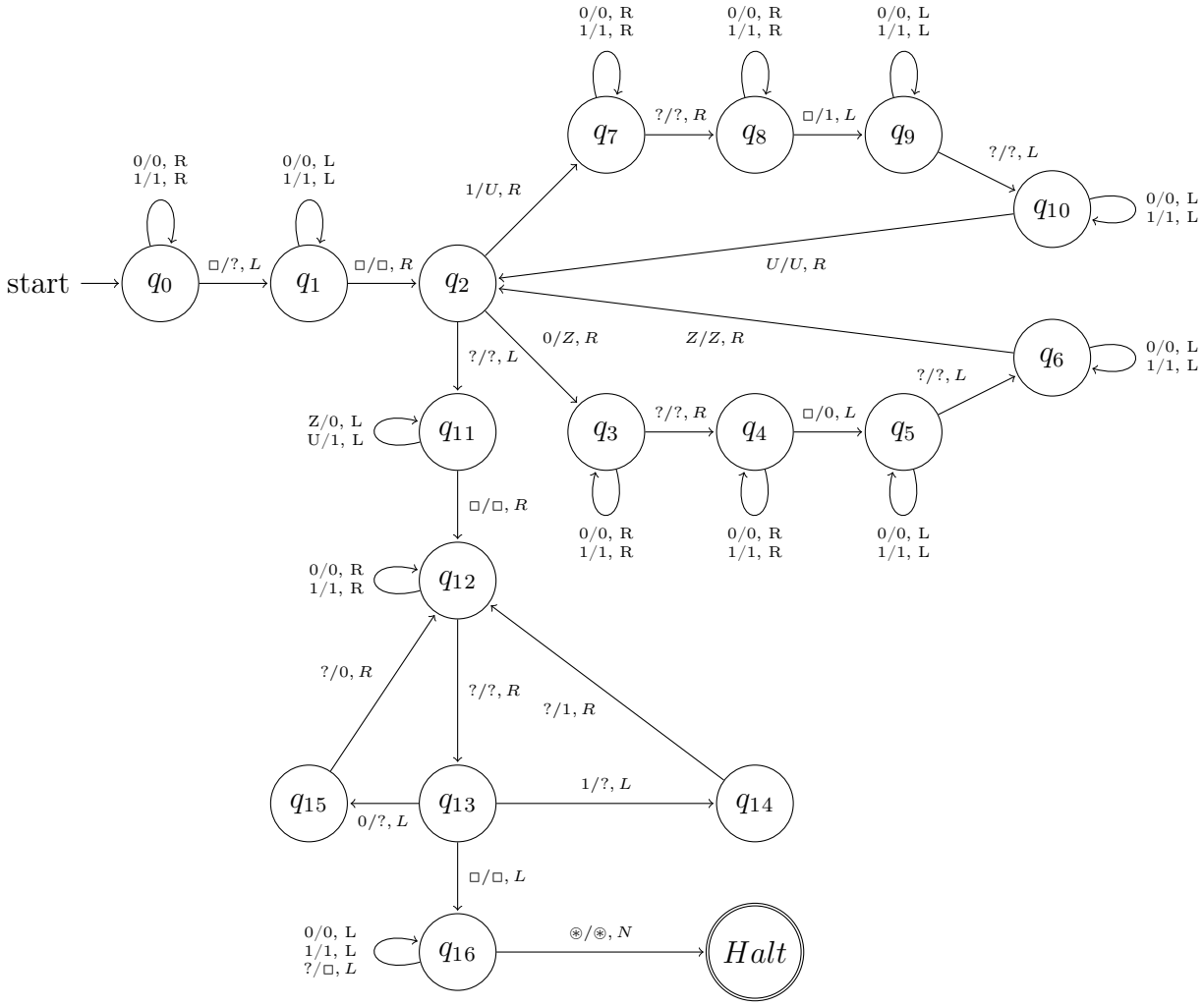
A∪B = {(C,W) | C é um AFD que aceita W não está contindo em A e B ao mesmo tempo}
O W não é aceito na MTc

2.c.

Sendo L decidível, pode-se sempre afirmar que L* é decidível também? Prove esta afirmação ou dê um contraexemplo.
Defini-se, para uma linguagem L qualquer, o seu fechamento, L*
Por exemplo:
Seja a linguagem L = 0, 11
L * = , 0, 11, 00, 011, 110, 1111, 000, 0011, 0110, ...
Se tanto L quanto L* são Turing-reconhecíveis, fazemos M1 ser o reconhecedor para L e M2 o reconhecedor para L*.
W = "aa"
M1 = ((q1,q2).(a),((q1,a,q2),(q2,a,q2)),q1,(q2))
M2 = ((q1,q2).(a,b),((q1,a,q2),(q1,b,q2),(q2,a,q2),(q2,b,q2)),q1,(q2))
O W aceito na M1, e também é aceito pelo M2
Provando que L* também é decidível

3. Exercício

Especifique uma Máquina de Turing (MT) que receba uma sting binária na fita e, ao final do processamento, duplique esta string. Por exemplo, suponha que a cadeia de entrada seja "⊛00110", ao final do processamento a máquina deve deixar a fica com "⊛0011000110". O caractere "⊛" está sendo usado com símbolo de início de fita e não deve ser duplicado.**(Legenda:** $\square = space$, $\circledast = start$, $? = aux$**)**



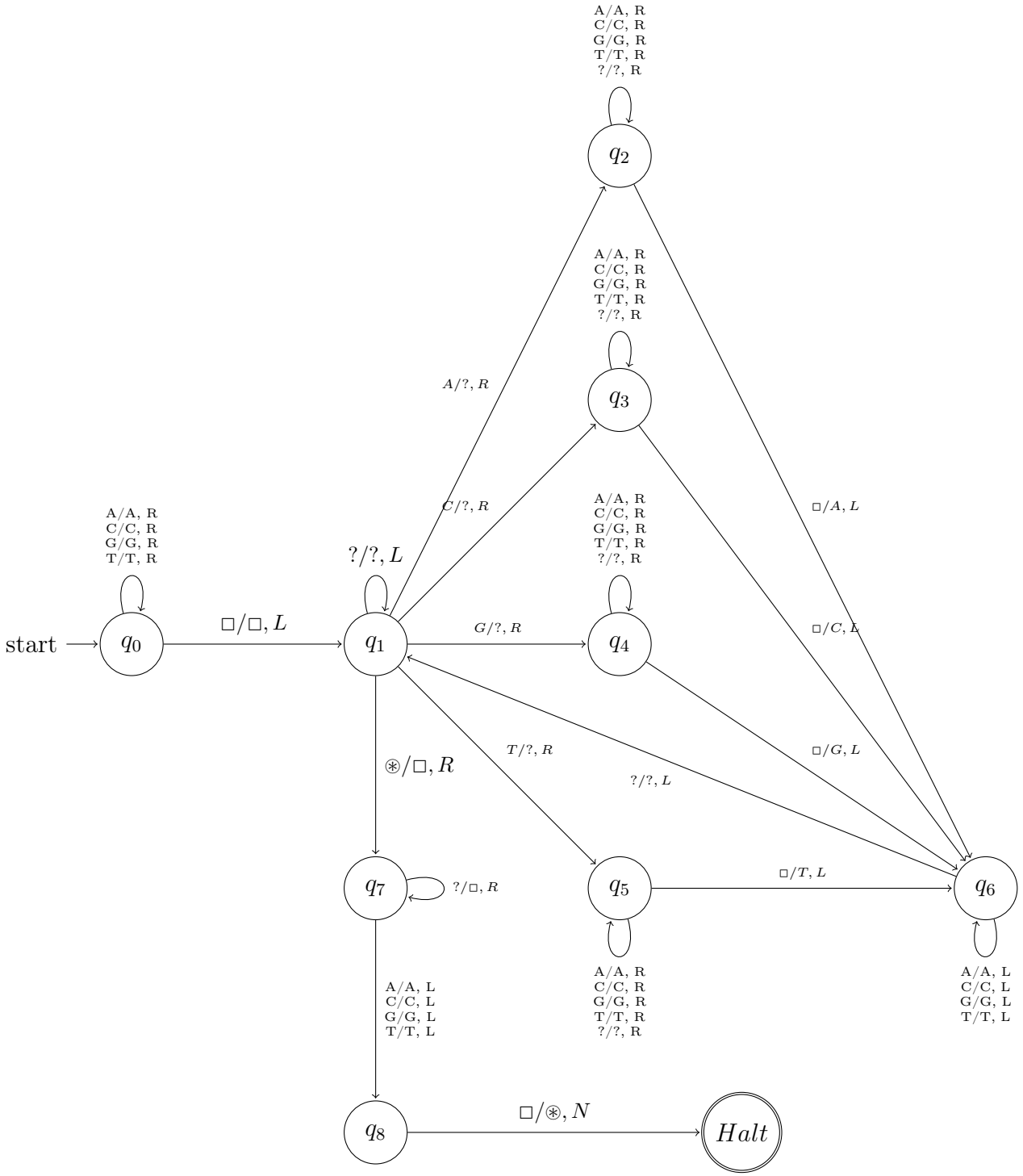

```
;Maquina de turing duplica binario
;http://morphett.info/turing/turing.html?a9e27fae74809ef7ce599afff5901fba

;Estado q0.
0 0 0 r 0
0 1 1 r 0
0 _ ? l 1

;Estado q1.
1 0 0 l 1
1 1 1 l 1
1 _ _ r 2

;Estado q2.
2 0 Z r 3
2 1 U r 7
2 ? ? l 11
```

```
;Estado q3.
3 0 0 r 3
3 1 1 r 3
3 ? ? r 4

;Estado q4.
4 0 0 r 4
4 1 1 r 4
4 _ 0 l 5

;Estado q5.
5 0 0 l 5
5 1 1 l 5
5 ? ? l 6

;Estado q6.
6 0 0 l 6
6 1 1 l 6
6 Z Z r 2

;Estado q7.
7 0 0 r 7
7 1 1 r 7
7 ? ? r 8

;Estado q8.
8 0 0 r 8
8 1 1 r 8
8 _ 1 l 9

;Estado q9.
9 0 0 l 9
9 1 1 l 9
9 ? ? l 10

;Estado q10.
10 0 0 l 10
10 1 1 l 10
10 U U r 2

;Estado q11.
11 U 1 l 11
11 Z 0 l 11
```

```
11 _ _ r 12

;Estado q12.
12 0 0 r 12
12 1 1 r 12
12 ? ? r 13

;Estado q13.
13 1 ? l 14
13 0 ? l 15
13 _ _ l 16

;Estado q14.
14 ? 1 r 12

;Estado q15.
15 ? 0 r 12

;Estado q16.
16 1 1 l 16
16 0 0 l 16
16 ? _ l 16
16 * * * halt-accept
```

4. Exercício

Especifique uma Máquina de Turing que reverta a sua entrada. Por exemplo, para a entrada "⊛GATTACA" a saída seria "⊛ACATTAG". Você pode assumir que a entrada é uma string genética. O caractere "⊛" está sendo usado com símbolo de início de fita e não deve ser revertido. **(Legenda:** $\square = space$, $\circledast = start$, $? = aux$**)**




```
;Maquina de turing para inverter uma string genética
http://morphett.info/turing/turing.html?aa8aa1d47ae552aab6882ee6d03d63b0

;Estado q0.
0 A A r 0
0 C C r 0
```

```
0 G G r 0
0 T T r 0
0 _ _ l 1


;Estado q1.
1 A ? r 2
1 C ? r 3
1 G ? r 4
1 T ? r 5
1 ? ? l 1
1 * _ r 7

;Estado q2.
2 A A r 2
2 C C r 2
2 G G r 2
2 T T r 2
2 ? ? r 2
2 _ A l 6

;Estado q3.
3 A A r 3
3 C C r 3
3 G G r 3
3 T T r 3
3 ? ? r 3
3 _ C l 6

;Estado q4.
4 A A r 4
4 C C r 4
4 G G r 4
4 T T r 4
4 ? ? r 4
4 _ G l 6

;Estado q5.
5 A A r 5
5 C C r 5
5 G G r 5
5 T T r 5
5 ? ? r 5
5 _ T l 6
```

```
;Estado q6.
6 A A l 6
6 C C l 6
6 G G l 6
6 T T l 6
6 ? ? l 1

;Estado q7.
7 A A l 8
7 C C l 8
7 G G l 8
7 T T l 8
7 ? _ r 7

;Estado q8.
8 _ * * halt-accept
```

5. Exercício

Escreva o programa1 applyBothTwice que receba como entrada uma única string S. O parâmetro S codifica três strings, P, Q e I utilizando a codificação ESS [1, pag. 61] da seguinte forma: S = ESS((P, Q), I). Sendo que P e Q são programas. A saída de applyBothTwice deve ser Q(P(Q(P(I)))).

```
import utils
from utils import rf
from universal import universal

I = 'yes'
S1 = utils.ESS(rf('P.py'), rf('Q.py'))

# String unica de entrada
S = utils.ESS(S1, I)


def applyBothTwice(S):
    (S1, I) = utils.DESS(S)
    (P, Q) = utils.DESS(S1)

    # Q(P(Q(P(I)))) formato de saida desejado
    x1 = universal(P, I)
    x2 = universal(Q, x1)
```

```
x3 = universal(P, x2)
x4 = universal(Q, x3)

return x4
```