

Máquinas de Turing

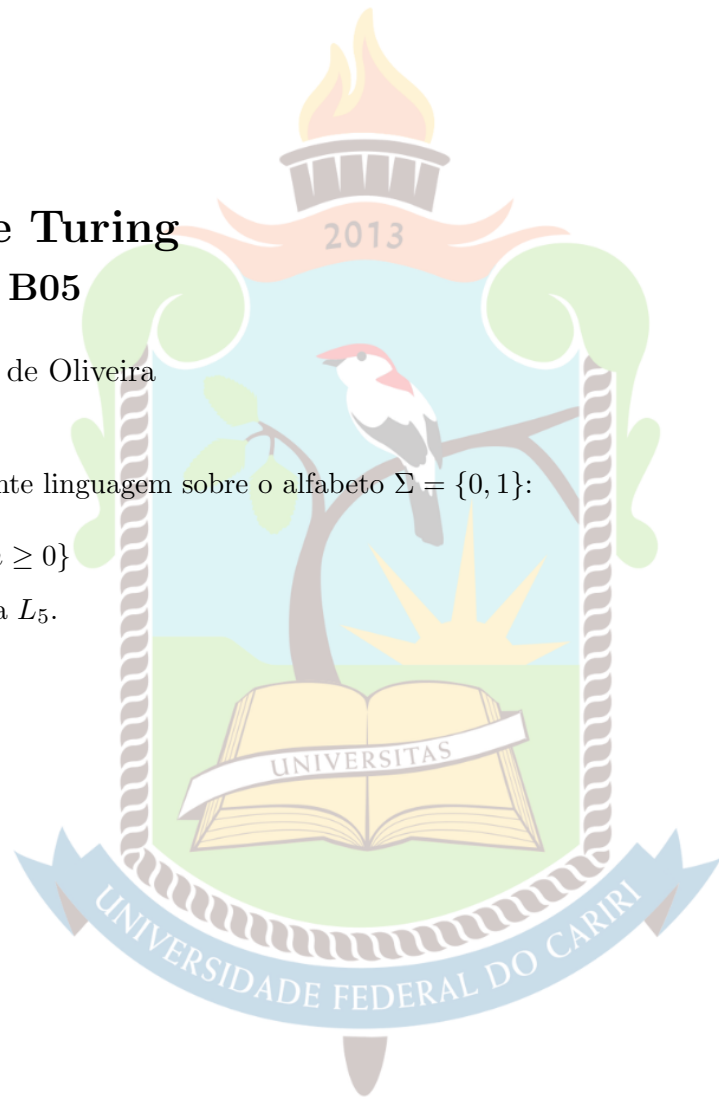
Exercício B05

Autor: Tony Esaú de Oliveira

Descrição do Problema: considere a seguinte linguagem sobre o alfabeto $\Sigma = \{0, 1\}$:

$$L_5 = \{0^n 1^n \mid n \geq 0\}$$

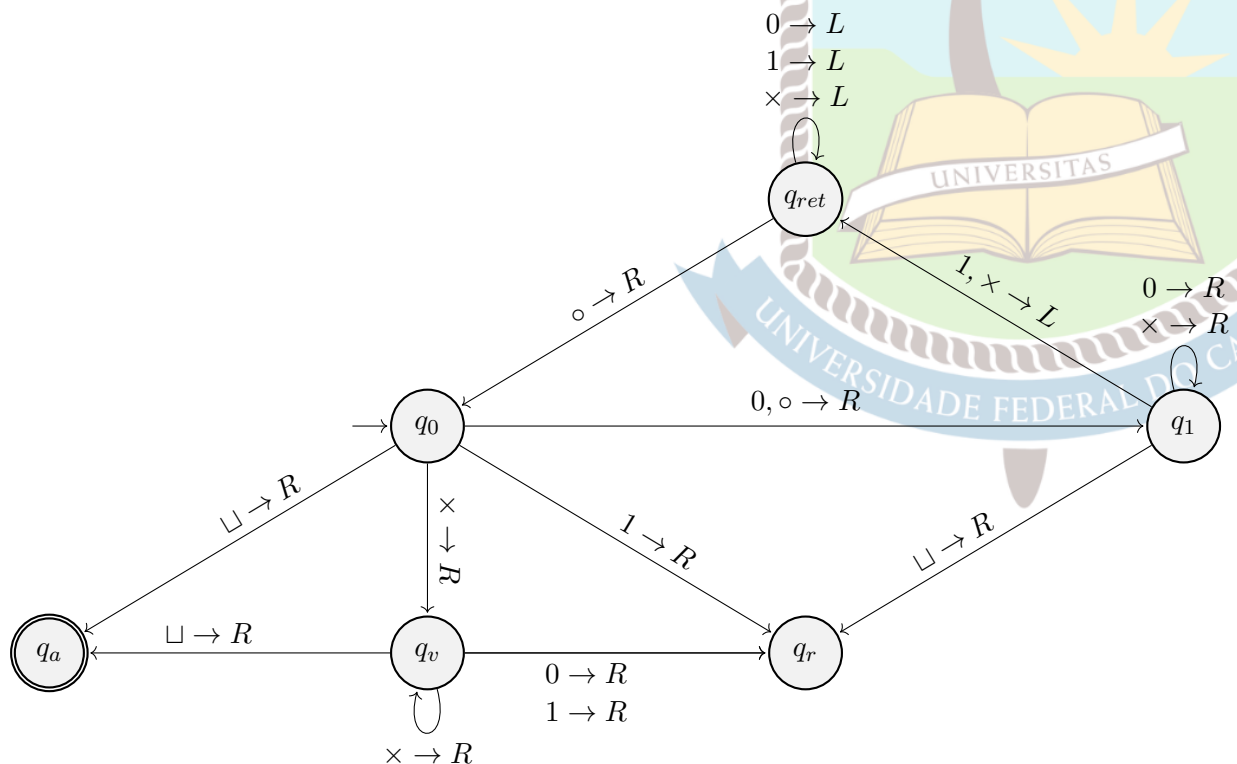
Implemente uma Máquina de Turing que decida L_5 .



Obs.: na próxima página haverá a resolução do problema. Por isso, antes de ver, tente implementar a máquina por si próprio.

Ideia de Resolução: a estratégia consiste em emparelhar cada 0 com um 1 correspondente à sua direita. O algoritmo marca cada 0 encontrado com o símbolo \circ , depois procura o primeiro 1 disponível à direita e o marca com \times , indicando que aquele par foi processado. Após marcar o par, a máquina retorna para o início da fita para buscar o próximo 0 não marcado e repetir o processo. Quando não há mais 0's para processar (encontra apenas símbolos \times ou chega ao final), a máquina entra em um estado de verificação que percorre a fita para garantir que não sobraram caracteres sem par. Se encontrar algum 0 ou 1 não marcado, rejeita; caso contrário, aceita a cadeia. A cadeia vazia é aceita diretamente, e cadeias que começam com 1 são rejeitadas imediatamente.

Diagrama Formal:



Código da máquina:

```
input: '000111001110'
blank: ' '
start state: q_0
```

table:

```
# Aceita cadeia vazia. Se o primeiro caractere for 1, rejeita.
# Se achar zero, marque com ◦ e prossiga para encontrar um 1.
# Se em algum momento encontrar ×, vá para o estado de verificação
verificar por caracteres restantes.
q_0:
  0: {write: ◦, R: q_1}
  1: {R: q_rejeita}
```

```

× : {R: q_verifica}
' ': {R: q_aceita}

# Estado que procura por 1's. Se achar vá para estado de retorno.
# Se bater no fim da fita, um 0 não pôde ser emparelhado, rejeita.
q_1:
  0: {R: q_1}
  1: {write: ×, L: q_retorna}
  × : {R: q_1}
  ' ': {R: q_rejeita}

# Estado auxiliar que retorna até o último caractere de 0 marcado com o.
q_retorna:
  0: {L: q_retorna}
  1: {L: q_retorna}
  o: {R: q_0}
  × : {L: q_retorna}

# Verifica caracteres restantes, se houver, rejeita. Caso contrário, aceita.
q_verifica:
  1: {R: q_rejeita}
  0: {R: q_rejeita}
  × : {R: q_verifica}
  ' ': {R: q_aceita}

q_aceita:

q_rejeita:

```

[Acesse o repositório aqui!](#)

