



Lógica & Computabilidade 2025-2

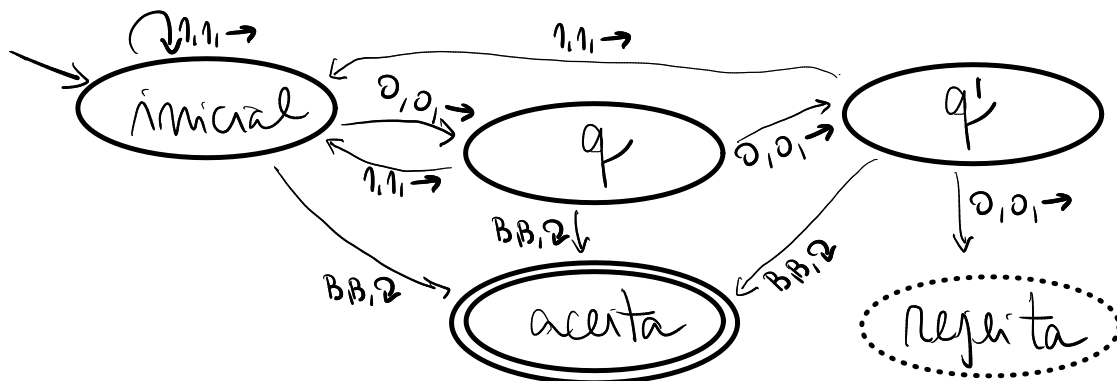
Prova 3

12 de dezembro de 2025

Você pode usar tudo que foi feito em sala ou listas de exercícios; apenas cite claramente quando o fizer. Você também pode usar uma questão da prova na solução de outra, desde que não crie dependências circulares.

Justifique todas as suas respostas!

Questão 1 (2 pontos). Descreva em linguagem clara para um ser humano a linguagem em $\{0,1\}^*$ que é aceita pela MT (lembrete: isso quer dizer “máquina de Turing”) dada pelo seguinte diagrama:



Questão 2 (2 pontos). Mostre que uma linguagem L é recursivamente enumerável (i.e., é a linguagem aceita por alguma MT) se, e somente se, L é reduzível à linguagem Aceita (lembrete: a linguagem Aceita é composta dos códigos de “MT M e entrada w ” tais que a execução de M em w termina aceitando).

Questão 3 (2 pontos). Seja $L \subseteq \Sigma^*$ uma linguagem decidível. Prove que L é reduzível a qualquer linguagem não-vazia que tenha complemento também não-vazio.

Questão 4.

Definição. Vamos chamar uma propriedade de linguagens de *interessante* se

- pelo menos uma MT aceita uma linguagem que tem essa propriedade, e
- pelo menos uma MT aceita uma linguagem que **não** tem essa propriedade.

Por exemplo, as propriedades “ser infinita”, “ser vazia” são interessantes, mas “ser uma linguagem” não é.

Teorema (de Rice). *Para qualquer propriedade P que seja interessante temos que*

$$L_P = \{c \in \Sigma_{\text{MT}}^* \mid c \text{ é código de uma MT cuja linguagem aceita tem a propriedade } P\}$$

é indecidível.

a (2 pontos). Vamos provar o Teorema de Rice.

Suponha que a linguagem vazia $\emptyset$ tenha a propriedade interessante P dada (o outro caso é análogo e você não precisa fazer, pode considerar como provado).

Seja N uma máquina de Turing cuja linguagem aceita não tem a propriedade P (note que N existe pois P é interessante).

Dado código c de “MT M e entrada w ”, seja $f(c)$ o código da MT M_w cujo funcionamento é:

1. Dada entrada x , primeiramente simulamos a execução de M em w .
 - Enquanto a simulação não terminar, continuamos simulando.
 - Se a simulação parar, seguimos pro próximo passo.
2. Simulamos N em x e copiamos seu comportamento, isto é:
 - Se essa simulação terminar aceitando, terminamos a nossa execução aceitando.
 - Se essa simulação terminar rejeitando ou por crash, terminamos a nossa execução rejeitando.
 - Enquanto a simulação não terminar, continuamos simulando.

Prove que f é uma redução da linguagem Parada ao complemento de L_P e explique por que isso implica que L_P é indecidível. (Lembrete: a linguagem Parada é composta dos códigos de “MT M e entrada w ” tais que a execução de M em w termina)

b (2 pontos). Prove que as linguagens abaixo são indecidíveis:

$$L_0 = \{c \in \Sigma_{\text{MT}}^* \mid c \text{ é código de MT que aceita infinitas palavras}\}$$

$$L_1 = \{c \in \Sigma_{\text{MT}}^* \mid c \text{ é código de MT que aceita alguma palavra de comprimento 3}\}$$

$$L_2 = \{c \in \Sigma_{\text{MT}}^* \mid c \text{ é código de MT que aceita a palavra “palavra”}\}$$