Instituto Superior Técnico

Lic. Engenharia Informática e de Computadores (Alameda)

Teoria da Computação

29 de Maio de 2019 | Teste 3A | Duração: 1h

Grupo I (5+5 valores)

Considere a linguagem

$$A = \{x\$y : \text{ existe } 1 < d < y \text{ tal que } d \text{ é divisor de } x\}$$

onde se assume que $x, y \in \{1\}^*$ são números naturais representados em unário.

a) Demonstre que $A \in \mathbf{NP}$.

b) A partir de **a)**, mostre que se $L_1 \in \mathbf{P}$ e $L_2 \leq_P A$ então $L_2 \setminus L_1 \in \mathbf{NP}$.

Grupo II (5 valores)

Seja Σ um alfabeto e $L \subseteq \Sigma^*$ uma linguagem.

Mostre que L é $\mathbf{P}$-difícil se e só se $L \neq \emptyset$ e $L \neq \Sigma^*$.

Grupo III (5 valores)

Enuncie o teorema de Savitch, e use-o para demonstrar que

$$\mathbf{P} \subseteq \mathbf{NP} \subseteq \mathbf{PSPACE} \subseteq \mathbf{EXPTIME}.$$

Instituto Superior Técnico

Lic. Engenharia Informática e de Computadores (Alameda)

Teoria da Computação

29 de Maio de 2019 Teste 3B Duração: 1h

Grupo I (5+5 valores)

Considere a linguagem $B \subseteq \{1, \#\}^*$ formada pelas palavras da forma

$$\#1^a\#1^b$$

para as quais existe $c \in \mathbb{N}$ tal que $1 < c < a$ e b é múltiplo de c.

a) Demonstre que $B \in \mathbf{NP}$.

b) A partir de **a)**, mostre que se $L_1 \leq_P B$ e $L_2 \in \mathbf{P}$ então $L_1 \cup \overline{L_2} \in \mathbf{NP}$.

Grupo II (5 valores)

Uma linguagem L diz-se *trivial* se $L = \emptyset$ ou $\overline{L} = \emptyset$.

Mostre que L é $\mathbf{P}$-completa se e só se $L \in \mathbf{P}$ e L não é trivial.

Grupo III (5 valores)

Enuncie o teorema de Savitch, e use-o para demonstrar que

$$\mathbf{P} \subseteq \mathbf{NP} \subseteq \mathbf{PSPACE} \subseteq \mathbf{EXPTIME}.$$

Instituto Superior Técnico

Lic. Engenharia Informática e de Computadores (Alameda)

Teoria da Computação

29 de Maio de 2019 | Teste 3C | Duração: 1h

Grupo I (5+5 valores)

Considere a linguagem

$$C = \{1^k\#1^m : \text{ existe } d \in \mathbb{N} \text{ com } 1 < d < k \text{ tal que } d \text{ divide } m\}.$$

a) Demonstre que $C \in \mathbf{NP}$.

b) A partir de **a)**, mostre que se $L_1 \in \mathbf{P}$ e $L_2 \leq_P C$ então $\overline{L_1} \cap L_2 \in \mathbf{NP}$.

Grupo II (5 valores)

Seja Σ um alfabeto. Uma linguagem $L \subseteq \Sigma^*$ diz-se *não-trivial* se $L \neq \Sigma^*$ e $\overline{L} \neq \Sigma^*$.

Mostre que L é **P**-difícil se e só se L é não-trivial.

Grupo III (5 valores)

Enuncie o teorema de Savitch, e use-o para demonstrar que

$$\mathbf{P} \subseteq \mathbf{NP} \subseteq \mathbf{PSPACE} \subseteq \mathbf{EXPTIME}.$$

Instituto Superior Técnico
Lic. Engenharia Informática e de Computadores (Alameda)

Teoria da Computação

29 de Maio de 2019 | Teste 3D | Duração: 1h

Grupo I (5+5 valores)

Considere a linguagem $D \subseteq \{\$, 1\}^*$ formada pelas palavras da forma

$$\$1^x\$1^y$$

para as quais existe $z \in \mathbb{N}$ tal que x é múltiplo de z e $1 < z < y$.

a) Demonstre que $D \in \mathbf{NP}$.

b) A partir de **a)**, mostre que se $L_1 \leq_P D$ e $L_2 \in \mathbf{P}$ então $L_1 \setminus \overline{L_2} \in \mathbf{NP}$.

Grupo II (5 valores)

Seja Σ um alfabeto e $L \subseteq \Sigma^*$ uma linguagem.

Mostre que L é $\mathbf{P}$-completa se e só se $L \in \mathbf{P}$ e $\emptyset \neq L \neq \Sigma^*$.

Grupo III (5 valores)

Enuncie o teorema de Savitch, e use-o para demonstrar que

$$\mathbf{P} \subseteq \mathbf{NP} \subseteq \mathbf{PSPACE} \subseteq \mathbf{EXPTIME}.$$

Instituto Superior Técnico

Lic. Engenharia Informática e de Computadores (Alameda)

Teoria da Computação

29 de Maio de 2019 Teste 3A - Resolução Duração: 1h

Grupo I (5+5 valores)

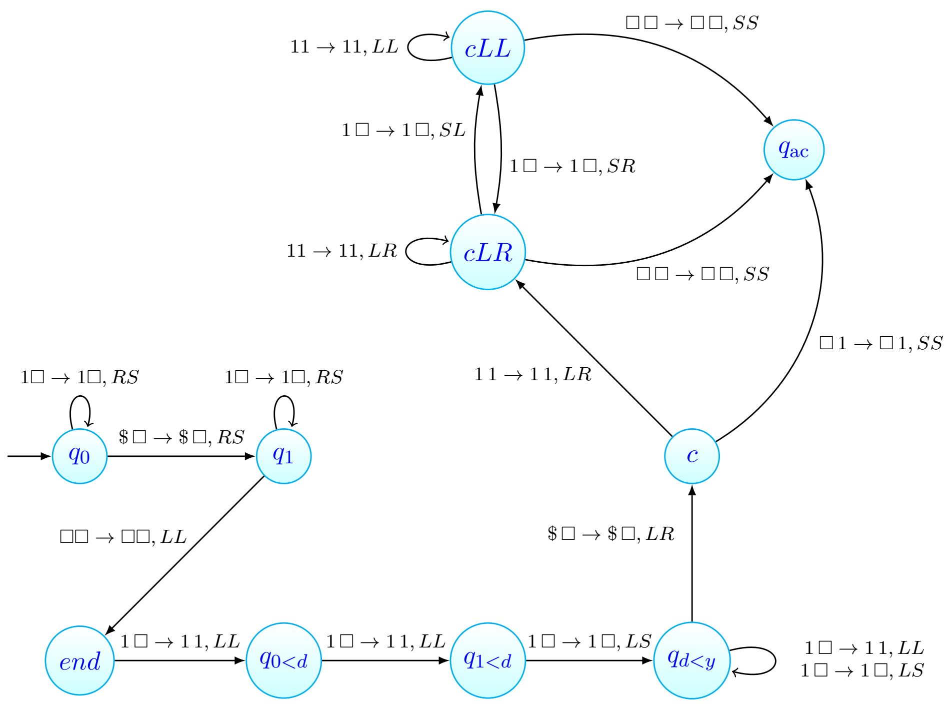
Considere a linguagem

$$A = \{x\$y : \text{ existe } 1 < d < y \text{ tal que } d \text{ é divisor de } x\}$$

onde se assume que $x, y \in \{1\}^*$ são números naturais representados em unário.

a) Demonstre que $A \in \mathbf{NP}$.

Resolução: Considere-se a máquina de Turing não-determinista N com alfabeto $\Gamma = \{1, \$, \square\}$ e duas fitas bidireccionais, representada por:



A máquina N decide A: (1) começa por percorrer o *input* confirmando que é da forma correcta $x\$y$ com $x, y \in \{1\}^*$ (estados q_0, q_1, end); (2) de seguida percorre y da direita para a esquerda em simultâneo escolhendo não-deterministicamente $1 < d < y$, que vai escrevendo na segunda fita (estados $q_{0<d}, q_{1<d}, q_{d<y}$); (3) finalmente verifica se d é divisor de x (estados c, cLR, cLL).

Em termos de eficiência, somando os tempos correspondentes a cada uma das 3 fases acima descritas, temos:

$$ntime_N(n) = \mathcal{O}(n) + \mathcal{O}(n) + \mathcal{O}(n) = \mathcal{O}(n).$$

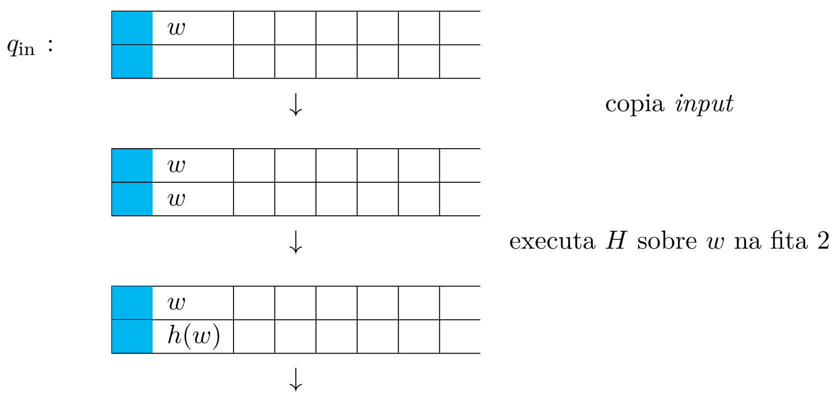
Sabemos que existe uma máquina com uma só fita de memória N_A equivalente a N com custo quadrático, isto é, $time_{N_A}(n) = \mathcal{O}(time_N(n)^2) = \mathcal{O}(n^2)$, pelo que podemos concluir que $A \in \mathbf{NP}$.

b) A partir de **a)**, mostre que se $L_1 \in \mathbf{P}$ e $L_2 \leq_P A$ então $L_2 \setminus L_1 \in \mathbf{NP}$.

Resolução: Da alínea **a)** temos uma máquina não-determinística N_A, de tempo quadrático, que decide A.

Se sabemos que $L_1 \in \mathbf{P}$ então existe uma máquina determinista D_1 que decide L_1 com $time_{D_1}(n) = \mathcal{O}(n^a)$ para $a \in \mathbb{N}$.

Se, adicionalmente, sabemos que $L_2 \leq_P A$ então existe uma função total h computável em tempo polinomial tal que $w \in L_2$ se e só se $h(w) \in A$. Assuma-se que h é calculada pela máquina H com $time_H(n) = \mathcal{O}(n^b)$ para $b \in \mathbb{N}$. Considere-se então a máquina não-determinística T, com 2 fitas, descrita por:



executa D_1 sobre w na fita 1

$\downarrow$ se rejeita

executa N_A sobre $h(w)$ na fita 2

$\downarrow$ se aceita

q_{ac}

A máquina T é classificadora pois h é total e D_1, N_A são classificadoras.

Facilmente, se T aceita o *input* w é porque D_1 rejeita w e N_A aceita $h(w)$. Isto significa que $w \notin L_1$ e que $h(w) \in A$, ou equivalentemente, que $w \in L_2$. Logo, tem-se $w \in L_2 \setminus L_1$.

Por outro lado, se $w \in L_2 \setminus L_1$ então $w \in L_2$ e $w \notin L_1$. Como $w \notin L_1$, necessariamente D_1 rejeita w. Tendo-se $w \in L_2$ tem-se também que $h(w) \in A$ e é certo que existe uma computação de N_A que aceita $h(w)$. Conclui-se que existe uma computação de T que aceita w.

Em termos de eficiência, tem-se

$$\begin{aligned} ntime_T(n) &\leq \mathcal{O}(n) + time_H(n) + time_{D_1}(n) + time_{N_A}(n + time_H(n)) \\ &\leq \mathcal{O}(n + n^b + n^a + (n + n^b)^2). \end{aligned}$$

A máquina T é equivalente a uma máquina de uma só fita com uma desaceleração quadrática, ainda polinomial, pelo que $L_2 \setminus L_1 \in \mathbf{NP}$.

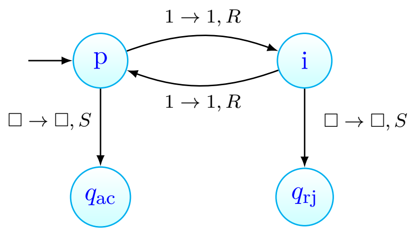
Grupo II (5 valores)

Seja Σ um alfabeto e $L \subseteq \Sigma^*$ uma linguagem.

Mostre que L é $\mathbf{P}$-difícil se e só se $L \neq \emptyset$ e $L \neq \Sigma^*$.

Resolução:

$(\Rightarrow)$: Considere-se a linguagem $A = \{1^n : n \text{ é par}\} \subseteq \{1\}^*$. Facilmente, tem-se que $A \in \mathbf{P}$ pois a máquina D_A descrita abaixo decide A e $time_{D_A}(n) = \mathcal{O}(n)$.

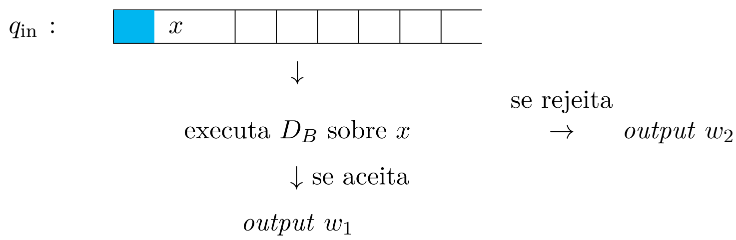


Se L é $\mathbf{P}$-difícil então tem-se $A \leq_P L$. Seja $f : \{1\}^* \to \Sigma^*$ a função total computável em tempo polinomial tal que $x \in A$ se e só se $f(x) \in L$. Obviamente, tem-se $11 \in A$ e consequentemente $f(11) \in L$ pelo que $L \neq \emptyset$. Analogamente, tem-se $1 \notin A$ e consequentemente $f(1) \notin L$ pelo que $L \neq \Sigma^*$.

$(\Leftarrow)$: Reciprocamente, se $L \neq \emptyset$ e $L \neq \Sigma^*$ tomem-se palavras $w_1 \in L$ e $w_2 \in \overline{L}$. Dada uma linguagem $B \in \mathbf{P}$ tem-se que $B \leq_P L$ com a redução definida pela função total

$$g(x) = \begin{cases} w_1 & \text{se } x \in B \\ w_2 & \text{se } x \notin B \end{cases}.$$

Se D_B é uma máquina que decide B com $time_{D_B}(n) = \mathcal{O}(n^b)$ para $b \in \mathbb{N}$, então g é calculada pela máquina G definida por



com $time_G(n) = time_{D_B}(n) + \mathcal{O}(space_{D_B}(n)) + \mathcal{O}(1) \leq \mathcal{O}(n^b + n^b + 1) = \mathcal{O}(n^b)$, correspondendo ao tempo necessário para executar D_B sobre o *input*, mais o tempo necessário para limpar a fita, mais o tempo (constante) necessário para escrever o *output*. Isto garante que g é computável em tempo polinomial.

Grupo III (5 valores)

Enuncie o teorema de Savitch, e use-o para demonstrar que

$$\mathbf{P} \subseteq \mathbf{NP} \subseteq \mathbf{PSPACE} \subseteq \mathbf{EXPTIME}.$$

Resolução:
O teorema de Savitch garante que se $n \leq f(n)$ então tem-se

$$\mathbf{NSPACE}(f(n)) \subseteq \mathbf{SPACE}(f(n)^2).$$

Demonstrem-se as inclusões.

Se $L \in \mathbf{P}$ então existe uma máquina de Turing determinista D que decide L com $time_D(n) = \mathcal{O}(n^a)$ para algum $a \in \mathbb{N}$. Mas as máquinas deterministas são um caso particular de máquina não-determinista pelo que D é também uma máquina não-determinista com $ntime_D(n) = time_D(n) = \mathcal{O}(n^a)$ e portanto $L \in \mathbf{NP}$. Concluimos que $\mathbf{P} \subseteq \mathbf{NP}$.

Se $L \in \mathbf{NP}$ então existe uma máquina de Turing não-determinista N que decide L com $ntime_N(n) = \mathcal{O}(n^a)$ para algum $a \in \mathbb{N}$. Mas sabemos que $nspace_N(n) \leq ntime_N(n)$ pois cada passo de computação pode visitar, no máximo, uma nova célula de memória. Portanto, temos que $nspace_N(n) = \mathcal{O}(n^a)$ e $L \in \mathbf{NSPACE}(n^a)$. Como $n \leq n^a$, o teorema de Savitch garante que $L \in \mathbf{SPACE}(n^{2a}) \subseteq \mathbf{PSPACE}$. Concluimos que $\mathbf{NP} \subseteq \mathbf{PSPACE}$.

Se $L \in \mathbf{PSPACE}$ então existe uma máquina de Turing determinista D que decide L com $space_D(n) = \mathcal{O}(n^a)$ para algum $a \in \mathbb{N}$. Mas para um classificador determinista D com conjunto de estados Q e alfabeto de trabalho Γ, em espaço limitado e sem entrar em ciclo, o comprimento máximo de uma computação (que termina) é o número de possíveis configurações distintas da máquina. Tem-se então que $time_D(n) \leq |Q|.space_D(n).|\Gamma|^{space_D(n)} = \mathcal{O}(|\Gamma|^{space_D(n)}) = 2^{\mathcal{O}(n^a)}$ e portanto $L \in \mathbf{EXPTIME}$. Concluimos que $\mathbf{PSPACE} \subseteq \mathbf{EXPTIME}$.