

Teoria da Computação

Nome: _____

Número: _____

Segundo Semestre 2018/2019

Mini-teste 1 (Versão C)

03/04/2019

Duração: 45 Minutos

Classificar (Sim/Não) _____

Quem não pretender ter nota nesta prova (ou seja, pretender “desistir”) deve indicar em cima que não pretende a prova classificada.

Este enunciado tem 5 páginas (incluindo esta). Apenas volte a página quando o professor assim o disser. Não é permitida a divulgação deste enunciado. A cópia em papel fornecida na prova deverá ficar sempre com um docente depois desta ser realizada (quer esteja preenchido ou não).

A folha de respostas múltiplas está anexa a este enunciado. Qualquer pergunta errada desconta 1/3 do seu valor no total da pontuação obtida com as respostas certas. Não é permitido o uso de qualquer tipo de material auxiliar ou electrónico enquanto estiver na sala em que decorre a prova.

Tabela de Pontuação

Question	Points	Score
1	10	
2	10	
3	10	
4	10	
5	10	
6	10	
7	10	
8	10	
9	10	
10	10	
Total:	100	

1. (10 points) Seja $\mathcal{A} = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$. A definição indutiva correcta do conjunto $\mathcal{N}$ das sequências não vazias de algarismos, é:

- A. $a \in \mathcal{A} \rightarrow a \in \mathcal{N}$ e $(a \in \mathcal{A} \wedge n \in \mathcal{N}) \rightarrow na \in \mathcal{N}$
- B. $a \in \mathcal{N}$ e $(a \in \mathcal{A} \wedge n \in \mathcal{N}) \rightarrow na \in \mathcal{N}$
- C. $a \in \mathcal{A} \rightarrow a \in \mathcal{N}$ e $(a \in \mathcal{A} \vee n \in \mathcal{N}) \rightarrow na \in \mathcal{N}$
- D. $a \in \mathcal{A} \rightarrow a \in \mathcal{N}$ e $na \in \mathcal{N}$
- E. Nenhuma das Anteriores

2. (10 points) Considere os conjuntos $\mathcal{N}$ e $\mathcal{A}$ definidos em cima e considere $n \in \mathcal{N}$ e $a, b \in \mathcal{A}$. O símbolo ε denota a sequência vazia.

A definição indutiva correcta da função rm sobre sequências não vazias de algarismos que remove o algarismo mais à direita de uma sequência não vazia, devolvendo uma sequência não vazia, é:

- A. $rm(a) = \varepsilon$ e $rm(na) = a$
- B. $rm(ab) = a$ e $rm(na) = n$
- C. $rm(a) = \varepsilon$ e $rm(na) = n$
- D. $rm(ab) = a$ e $rm(na) = a$
- E. Nenhuma das Anteriores

3. (10 points) Considere que φ, ψ e δ são fórmulas de primeira ordem. Qual das seguintes fórmulas é de primeira ordem?

- A. $\exists \top (((\psi \rightarrow \perp) \vee (\perp \rightarrow \perp)) \rightarrow ((\psi \rightarrow \perp) \vee (\perp \rightarrow \perp)))$
- B. $\exists \varphi ((\varphi \rightarrow \perp) \rightarrow (\psi \vee \delta))$
- C. $\exists \psi (((\varphi \vee \psi) \rightarrow \perp) \rightarrow (\varphi \rightarrow \perp))$
- D. $\exists \perp ((\varphi \rightarrow \perp) \wedge \psi) \rightarrow \perp$
- E. Nenhuma das anteriores

4. (10 points) Pretende-se definir o conjunto Z de todos os números inteiros, respeitando o princípio da separação. A definição correspondente é (escolha a verdadeira):

- A. $Z = \{x \mid \exists y (y \in NAT \wedge x = -y)\}$
- B. $Z = \{x \in INT \mid x = -x\}$
- C. $Z = NAT \cup \{x \mid \exists y (y \in NAT \wedge x = -y)\}$
- D. $Z = \mathbb{N}_0 \cup \mathbb{N}_0$
- E. Nenhuma das Anteriores

5. (10 points) Qual das seguintes respostas corresponde à derivação correcta da fórmula de primeira ordem $(\text{id}(x) = \text{fulano}) \wedge \text{eCliente}(x)$ sabendo que a assinatura é tal que: $\text{fulano} \in SF_0$, $\text{id} \in SF_1$, $\text{eCliente} \in SP_1$, $= \in SP_2$ e $x \in X$.

A.

$$\frac{\frac{\frac{x \in X}{\text{id}(x) \in T_\Sigma^X} \text{ (FUN)} \quad \frac{\text{fulano} \in SF_0}{\text{fulano} \in T_\Sigma^X} \text{ (CONST)}}{(\text{id}(x) = \text{fulano}) \in F_\Sigma^X} \text{ (PRED)} \quad D}{(\text{id}(x) = \text{fulano}) \wedge \text{eCliente}(x) \in F_\Sigma^X} \text{ (CON)}$$

Sendo D

$$\frac{x \in X \quad \text{eCliente} \in SP_1}{\text{eCliente}(x) \in F_\Sigma^X} \text{ (PRED)}$$

B.

$$\frac{\frac{\frac{x \in X}{x \in T_\Sigma^X} \text{ (VAR)} \quad \frac{\text{id} \in SF_1}{\text{id}(x) \in T_\Sigma^X} \text{ (FUN)} \quad \frac{\text{fulano} \in SF_0}{\text{fulano} \in T_\Sigma^X} \text{ (CONST)}}{(\text{id}(x) = \text{fulano}) \in F_\Sigma^X} \text{ (PRED)} \quad D}{(\text{id}(x) = \text{fulano}) \wedge \text{eCliente}(x) \in F_\Sigma^X} \text{ (CON)}$$

Sendo D

$$\frac{\frac{x \in X}{x \in T_\Sigma^X} \text{ (VAR)} \quad \text{eCliente} \in SP_1}{\text{eCliente}(x) \in F_\Sigma^X} \text{ (PRED)}$$

C.

$$\frac{\frac{\frac{x \in X}{x \in T_\Sigma^X} \text{ (VAR)} \quad \frac{\text{id} \in SF_1}{\text{id}(x) \in T_\Sigma^X} \text{ (FUN)} \quad \frac{\text{fulano} \in SF_0}{\text{fulano} \in T_\Sigma^X} \text{ (CONST)}}{(\text{id}(x) = \text{fulano}) \in F_\Sigma^X} \text{ (PRED)} \quad = \in SP_2}{(\text{id}(x) = \text{fulano}) \wedge \text{eCliente}(x) \in F_\Sigma^X} \text{ (CON)}$$

Sendo D

$$\frac{\frac{x \in X}{x \in T_\Sigma^X} \text{ (VAR)} \quad \text{eCliente} \in SP_1}{\text{eCliente}(x) \in F_\Sigma^X} \text{ (PRED)}$$

D.

$$\frac{\frac{\frac{x \in X}{\text{id}(x) \in T_\Sigma^X} \text{ (FUN)} \quad \frac{\text{fulano} \in SF_0}{\text{fulano} \in T_\Sigma^X} \text{ (CONST)}}{(\text{id}(x) = \text{fulano}) \in F_\Sigma^X} \text{ (PRED)} \quad = \in SP_2}{(\text{id}(x) = \text{fulano}) \wedge \text{eCliente}(x) \in F_\Sigma^X} \text{ (CON)}$$

Sendo D

$$\frac{x \in X \quad \text{eCliente} \in SP_1}{\text{eCliente}(x) \in F_\Sigma^X} \text{ (PRED)}$$

E. Nenhuma das anteriores.

Uma loja online permite que os clientes registados definam listas de produtos. Cada cliente é identificado por um nome único e pode ter um número arbitrário de listas.

Uma lista tem um nome (único, nas listas do cliente a que pertence) e um conjunto de nomes de produtos. Inicialmente, o conjunto das listas do novo cliente é vazio.

Considere que o nome da loja, de um cliente, de cada lista e dos produtos são strings.

6. (10 points) A definição correcta do conjunto $LOJA$ de todas as lojas, constituídas por um nome único e um conjunto de clientes, é:

- A. $LOJA \stackrel{\text{def}}{=} NOME \times \wp(CLIENTE)$, sendo $CLIENTE \stackrel{\text{def}}{=} NOME \times LISTA$ e $LISTA \stackrel{\text{def}}{=} NOME \times \wp(PRODUTO)$
- B. $LOJA \stackrel{\text{def}}{=} NOME \times CLIENTE$, sendo $CLIENTE \stackrel{\text{def}}{=} NOME \times \wp(LISTA)$ e $LISTA \stackrel{\text{def}}{=} NOME \times \wp(PRODUTO)$
- C. $LOJA \stackrel{\text{def}}{=} NOME \times \wp(CLIENTE)$, sendo $CLIENTE \stackrel{\text{def}}{=} NOME \times \wp(LISTA)$ e $LISTA \stackrel{\text{def}}{=} NOME \times \wp(PRODUTO)$
- D. $LOJA \stackrel{\text{def}}{=} NOME \times \wp(CLIENTE)$, sendo $CLIENTE \stackrel{\text{def}}{=} NOME \times \wp(LISTA)$ e $LISTA \stackrel{\text{def}}{=} NOME \times PRODUTO$
- E. Nenhuma das anteriores.

7. (10 points) O predicado de primeira ordem que verifica se um cliente, identificado pelo seu nome, existe numa loja, é:

- A. $\text{existeCliente}(l, n) \stackrel{\text{def}}{=} \exists c (c \in \pi_2(l) \vee n = \pi_1(c))$
- B. $\text{existeCliente}(l, n) \stackrel{\text{def}}{=} \forall c (c \in \pi_2(l) \vee n = \pi_1(c))$
- C. $\text{existeCliente}(l, n) \stackrel{\text{def}}{=} \exists c (c \in \pi_2(l) \rightarrow n = \pi_1(c))$
- D. $\text{existeCliente}(l, n) \stackrel{\text{def}}{=} \forall c (c \in \pi_2(l) \wedge n = \pi_1(c))$
- E. Nenhuma das anteriores.

8. (10 points) A função que verifica se um cliente, identificado pelo seu nome, existe numa loja, é:

- A. $\text{eCliente} \in NOME \times LOJA \rightarrow \{\top, \perp\}$
 $\text{eCliente} \stackrel{\text{def}}{=} \{(n, l) \mapsto b \mid \text{existeCliente}(l, n) \rightarrow b = TRUE \wedge \neg \text{existeCliente}(l, n) \rightarrow b = FALSE\}$
- B. $\text{eCliente} \in NOME \times LOJA \rightarrow \{\top, \perp\}$
 $\text{eCliente} \stackrel{\text{def}}{=} \{(n, l) \mapsto b \mid \text{existeCliente}(l, n) \leftrightarrow b = TRUE\}$
- C. $\text{eCliente} \in NOME \times LOJA \rightarrow BOOL$
 $\text{eCliente} \stackrel{\text{def}}{=} \{(n, l) \mapsto b \mid \text{existeCliente}(l, n) \rightarrow b = TRUE \wedge \neg \text{existeCliente}(l, n) \rightarrow b = FALSE\}$

- D. $\text{eCliente} \in NOME \times LOJA \rightarrow BOOL$
 $\text{eCliente} \stackrel{\text{def}}{=} \{(n, l) \mapsto b \mid b = \text{existeCliente}(l, n)\}$
- E. Nenhuma das anteriores.
9. (10 points) A função que remove um cliente, identificado pelo seu nome, numa loja, é:
- A. $\text{remCliente} \in NOME \times LOJA \rightarrow LOJA$
 $\text{remCliente} \stackrel{\text{def}}{=} \{(n, l) \mapsto l' \mid \exists c (\pi_1(c) = n \wedge \text{eCliente}(n, l) = TRUE \wedge l' = (\pi_1(l), \pi_2(l) \setminus \{c\}))\}$
- B. $\text{remCliente} \in NOME \times LOJA \rightarrow LOJA$
 $\text{remCliente} \stackrel{\text{def}}{=} \{(n, l) \mapsto l' \mid \exists c (\pi_1(c) = n \wedge \text{eCliente}(n, l) \wedge l' = (\pi_1(l), \pi_2(l) \setminus \{c\}))\}$
- C. $\text{remCliente} \in NOME \times LOJA \rightarrow LOJA$
 $\text{remCliente} \stackrel{\text{def}}{=} \{(n, l) \mapsto l' \mid \exists c (l' = (\pi_1(l), \pi_2(l) \setminus \{c\}))\}$
- D. $\text{remCliente} \in NOME \times LOJA \rightarrow LOJA$
 $\text{remCliente} \stackrel{\text{def}}{=} \{(n, l) \mapsto l' \mid \pi_1(c) = n \wedge \text{eCliente}(n, l) = TRUE \wedge l' = (\pi_1(l), \pi_2(l) \setminus \{c\})\}$
- E. Nenhuma das anteriores.
10. (10 points) A função que adiciona uma nova lista a um cliente, identificado pelo seu nome, numa loja, é:
- A. $\text{adCliente} \in NOME \times NOME \times LOJA \rightarrow LOJA$
 $\text{adCliente} \stackrel{\text{def}}{=} \{(n_c, n_l, s) \mapsto s' \mid \exists c (c \in \pi_2(s) \wedge n_c = \pi_1(c) \wedge \neg \text{temLista}(c, n_l) \wedge s' = (\pi_1(s), \pi_2(s) \setminus \{c\} \cup \{(n_l, \emptyset)\}))\}$
com $\text{temLista}(c, n_l) = \exists l (l \in \pi_2(c) \wedge n_l = \pi_1(l))$
- B. $\text{adCliente} \in NOME \times NOME \times LOJA \rightarrow LOJA$
 $\text{adCliente} \stackrel{\text{def}}{=} \{(n_c, n_l, s) \mapsto s' \mid \exists c (c \in \pi_2(s) \wedge n_c = \pi_1(c) \wedge \neg \text{temLista}(c, n_l) \wedge s' = (\pi_1(c), \pi_2(c) \cup \{(n_l, \emptyset)\}))\}$
com $\text{temLista}(c, n_l) = \exists l (l \in \pi_2(c) \wedge n_l = \pi_1(l))$
- C. $\text{adCliente} \in NOME \times NOME \times LOJA \rightarrow LOJA$
 $\text{adCliente} \stackrel{\text{def}}{=} \{(n_c, n_l, s) \mapsto s' \mid \exists c (c \in \pi_2(s) \vee n_c = \pi_1(c) \vee \neg \text{temLista}(c, n_l) \vee s' = (\pi_1(s), \pi_2(s) \setminus \{c\} \cup \{(\pi_1(c), \pi_2(c) \cup \{(n_l, \emptyset)\})\}))\}$
com $\text{temLista}(c, n_l) = \exists l (l \in \pi_2(c) \wedge n_l = \pi_1(l))$
- D. $\text{adCliente} \in NOME \times NOME \times LOJA \rightarrow LOJA$
 $\text{adCliente} \stackrel{\text{def}}{=} \{(n_c, n_l, s) \mapsto s' \mid \exists c (c \in \pi_2(s) \wedge n_c = \pi_1(c) \wedge \neg \text{temLista}(c, n_l) \wedge s' = (\pi_1(s), \pi_2(s) \setminus \{c\} \cup \{(\pi_1(c), \pi_2(c) \cup \{(n_l, \emptyset)\})\}))\}$
com $\text{temLista}(c, n_l) = \exists l (l \in \pi_2(c) \wedge n_l = \pi_1(l))$
- E. Nenhuma das anteriores.