

Teoria da Computação

Nome: _____

Número: _____

Segundo Semestre 2017/2018

Mini-teste 1 (Versão A)

27/3/2018

Duração: 45 Minutos

Classificar (Sim/Não) _____

Este enunciado tem 5 páginas (incluindo esta) e 10 questões.

Apenas voltar a página quando o professor assim o disser. 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.

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) Pretende-se definir o conjunto C de todos os números inteiros (de INT) ímpares menores que 10. A definição por compreensão correspondente é (escolha a verdadeira):

- A. $C = \{1, 3, 5, 7, 9\}$
B. $C = \{x \in INT \mid x < 10 \wedge x \% 2 = 1\}$
C. $C = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\} \cup \{x \in INT \mid x \% 2 = 1\}$
D. $C = \{x \in NAT \mid x < 10 \wedge x \% 2 = 1\}$
E. Nenhuma das Anteriores

2. (10 points) A definição indutiva correcta do conjunto $\mathcal{W}$ das palavras binárias não vazias, é:

- A. $0 \in \mathcal{W}, 1 \in \mathcal{W}, w \in \mathcal{W} \rightarrow 0w \in \mathcal{W}, \text{ e } w \in \mathcal{W} \rightarrow 1w \in \mathcal{W}$
B. $0 \in \mathcal{W} \text{ e } 1 \in \mathcal{W}$
C. $w \in \mathcal{W} \rightarrow 0w \in \mathcal{W} \text{ e } w \in \mathcal{W} \rightarrow 1w \in \mathcal{W}$
D. $0 \in \mathcal{W} \text{ ou } 1 \in \mathcal{W} \text{ ou } w \in \mathcal{W} \rightarrow 0w \in \mathcal{W} \text{ ou } w \in \mathcal{W} \rightarrow 1w \in \mathcal{W}$
E. $(0 \in \mathcal{W} \text{ ou } 1 \in \mathcal{W}) \text{ e } (w \in \mathcal{W} \rightarrow 0w \in \mathcal{W} \text{ ou } w \in \mathcal{W} \rightarrow 1w \in \mathcal{W})$

3. (10 points) Considere o conjunto $\mathcal{W}$ definido em cima e seja $w \in \mathcal{W}$. A definição indutiva correcta da função rm sobre palavras binárias não vazias que remove o bit mais à direita de uma palavra não vazia é:

- A. $rm(w0) = w \text{ ou } rm(w1) = w$
B. $rm(0) = 0, rm(1) = 1, rm(w0) = w, \text{ e } rm(w1) = w$
C. $rm(0) = \varepsilon, rm(1) = \varepsilon, rm(0w) = rm(w), \text{ e } rm(1w) = rm(w)$
D. $rm(0) = \varepsilon, rm(1) = \varepsilon, rm(0w) = w, \text{ e } rm(1w) = w$
E. $rm(0) = 0, rm(1) = 1, rm(w0) = rm(w), \text{ e } rm(w1) = rm(w)$

4. (10 points) Qual das seguintes expansões de abreviaturas está incorrecta?

- A. $\neg(\neg\varphi \wedge \psi) \stackrel{\text{abv}}{=} ((\varphi \rightarrow \perp) \wedge \psi) \rightarrow \perp$
B. $\neg\varphi \leftrightarrow (\psi \vee \delta) \stackrel{\text{abv}}{=} ((\varphi \rightarrow \perp) \rightarrow (\psi \vee \delta))$
C. $(\neg(\varphi \vee \psi) \rightarrow \neg\varphi) \stackrel{\text{abv}}{=} (((\varphi \vee \psi) \rightarrow \perp) \rightarrow (\varphi \rightarrow \perp))$
D. $((\neg\psi) \vee \top) \rightarrow ((\neg\psi) \vee \top) \stackrel{\text{abv}}{=} (((\psi \rightarrow \perp) \vee (\perp \rightarrow \perp)) \rightarrow ((\psi \rightarrow \perp) \vee (\perp \rightarrow \perp)))$
E. Nenhuma das anteriores

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

A.
$$\frac{x \in X \quad \frac{\frac{x \in T_\Sigma^X \quad name \in SP_1}{name(x) \in T_\Sigma^X} \text{ (FUN)} \quad User \in SP_1}{User(name(x)) \in F_\Sigma^X} \text{ (PRED)}}{\forall x User(name(x)) \in F_\Sigma^X} \text{ (UNIV)}$$

B.
$$\frac{x \in X \quad \frac{\frac{x \in X}{x \in T_\Sigma^X} \text{ (VAR)} \quad name \in SP_1}{name(x) \in T_\Sigma^X} \text{ (PRED)} \quad User \in SP_1}{User(name(x)) \in F_\Sigma^X} \text{ (PRED)}}{\forall x User(name(x)) \in F_\Sigma^X} \text{ (UNIV)}$$

C.
$$\frac{x \in X \quad \frac{\frac{x \in X}{x \in T_\Sigma^X} \text{ (VAR)} \quad name \in SP_1}{name(x) \in T_\Sigma^X} \text{ (FUN)} \quad User \in SP_1}{User(name(x)) \in F_\Sigma^X} \text{ (FUN)}}{\forall x User(name(x)) \in F_\Sigma^X} \text{ (UNIV)}$$

D.
$$\frac{x \in X \quad \frac{\frac{x \in X}{x \in T_\Sigma^X} \text{ (VAR)} \quad name \in SF_1}{name(x) \in T_\Sigma^X} \text{ (FUN)} \quad User \in SP_1}{User(name(x)) \in F_\Sigma^X} \text{ (PRED)}}{\forall x User(name(x)) \in F_\Sigma^X} \text{ (UNIV)}$$

E. Nenhuma das anteriores.

Considere um sistema de armazenamento de ficheiros na *cloud*. O sistema é composto por um conjunto global de pastas (o repositório) e um conjunto de utilizadores.

Uma pasta é identificada por um nome e contém uma colecção de ficheiros. Um ficheiro tem um nome e um conteúdo (que é um elemento do conjunto *TEXT*).

Cada utilizador é univocamente identificado por um nome e tem a sua colecção de pastas (locais). Cada pasta local é uma versão de uma pasta no repositório do sistema. Uma pasta local pode não coincidir com a sua versão no repositório do sistema, pois as pastas do utilizador podem não estar sincronizadas com o repositório (estarem desactualizadas ou conterem alterações ainda não colocadas no repositório).

Seja $NAME \stackrel{\text{def}}{=} STRING$ and $USER \stackrel{\text{def}}{=} STRING$.

6. (10 points) A definição correcta do conjunto *FOLDER* de todas as pastas é:

- A. $FILE \stackrel{\text{def}}{=} NAME \times NAME$
- B. $FILE \stackrel{\text{def}}{=} NAME$ e $FOLDER \stackrel{\text{def}}{=} NAME \times FILE$
- ☒ C. $FILE \stackrel{\text{def}}{=} NAME \times TEXT$ e $FOLDER \stackrel{\text{def}}{=} NAME \times \wp(FILE)$
- D. $FILE \stackrel{\text{def}}{=} NAME \times TEXT$ e $FOLDER \stackrel{\text{def}}{=} NAME \times FILE$
- E. $FILE \stackrel{\text{def}}{=} NAME$ e $FOLDER \stackrel{\text{def}}{=} NAME \times \wp(FILE)$

7. (10 points) A definição correcta do sistema *SDROP* é:

- ☒ A. $CLIENT \stackrel{\text{def}}{=} USER \times \wp(FOLDER)$ e $SDROP \stackrel{\text{def}}{=} \wp(FOLDER) \times \wp(CLIENT)$
- B. $CLIENT \stackrel{\text{def}}{=} USER \times FOLDER$ e $SDROP \stackrel{\text{def}}{=} \wp(FOLDER) \times \wp(CLIENT)$
- C. $CLIENT \stackrel{\text{def}}{=} USER \times \wp(FOLDER)$ e $SDROP \stackrel{\text{def}}{=} \wp(FOLDER) \times CLIENT$
- D. $CLIENT \stackrel{\text{def}}{=} \wp(FOLDER)$ e $SDROP \stackrel{\text{def}}{=} \wp(FOLDER) \times \wp(CLIENT)$
- E. $CLIENT \stackrel{\text{def}}{=} USER \times \wp(FOLDER)$ e $SDROP \stackrel{\text{def}}{=} \wp(FOLDER)$

8. (10 points) Considerando u e s variáveis, respectivamente representando um utilizador genérico e um sistema, um predicado de primeira ordem que verifica se um utilizador existe num sistema é:

- A. $\text{userExists}(u, s) \stackrel{\text{def}}{=} \forall c. c \in \pi_2(s) \wedge u = \pi_1(c)$
- B. $\text{userExists}(u, s) \stackrel{\text{def}}{=} c \in \pi_2(s) \wedge u = \pi_1(c)$
- C. $\text{userExists}(u, s) \stackrel{\text{def}}{=} \exists c. c \in \pi_2(s) \rightarrow u = \pi_1(c)$
- ☒ D. $\text{userExists}(u, s) \stackrel{\text{def}}{=} \exists c. c \in \pi_2(s) \wedge u = \pi_1(c)$
- E. $\text{userExists}(u, s) \stackrel{\text{def}}{=} \exists c. u = \pi_1(c)$

9. (10 points) A definição da função que verifica se um utilizador existe no sistema é:

A. $\text{hasuser} \in \text{SDROP} \rightarrow \text{BOOL}$

$$\text{hasuser} \stackrel{\text{def}}{=} \{(s, u) \mapsto b \mid b = \text{userExists}(u, s)\}$$

B. $\text{hasuser} \in \text{SDROP} \times \text{STRING} \rightarrow \text{BOOL}$

$$\text{hasuser} \stackrel{\text{def}}{=} \{(s, u) \mid b = \text{userExists}(u, s)\}$$

C. $\text{hasuser} \in \text{SDROP} \times \text{SDROP} \rightarrow \text{BOOL}$

$$\text{hasuser} \stackrel{\text{def}}{=} \{(s, u) \mapsto b \mid (\text{userExists}(u, s) \rightarrow b = \text{TRUE}) \wedge (\neg \text{userExists}(u, s) \rightarrow b = \text{FALSE})\}$$

D. $\text{hasuser} \in \text{SDROP} \times \text{STRING} \rightarrow \text{BOOL}$

$$\text{hasuser} \stackrel{\text{def}}{=} \{(s, u) \mapsto b \mid (\text{userExists}(u, s) \rightarrow b = \text{TRUE}) \wedge (\neg \text{userExists}(u, s) \rightarrow b = \text{FALSE})\}$$

E. $\text{hasuser} \in \text{SDROP} \times \text{STRING} \rightarrow \text{NAT}$

$$\text{hasuser} \stackrel{\text{def}}{=} \{(s, u) \mapsto b \mid (\text{userExists}(u, s) \rightarrow b = \text{TRUE}) \wedge (\neg \text{userExists}(u, s) \rightarrow b = \text{FALSE})\}$$

10. (10 points) Considere que quando se cria um novo utilizador, o seu nome não pode existir no sistema e o seu conjunto de pastas locais fica vazio.

A função $\text{addclient} \in \text{SDROP} \times \text{STRING} \rightarrow \text{SDROP}$ que define a criação de um novo utilizador do sistema é:

A. $\text{addclient} \stackrel{\text{def}}{=} \{(s, u) \mapsto s' \mid s' = (\pi_1(s), \pi_2(s) \cup \{(u, \emptyset)\}) \wedge \neg \text{hasUser}(s, u)\}$

B. $\text{addclient} \stackrel{\text{def}}{=} \{(s, u) \mapsto s' \mid s' = \pi_2(s) \cup \{(u, \emptyset)\} \wedge \neg \text{hasUser}(s, u)\}$

C. $\text{addclient} \stackrel{\text{def}}{=} \{(s, u) \mapsto s' \mid s' = (\pi_1(s), \pi_2(s) \subseteq \{(u, \emptyset)\}) \wedge \neg \text{hasUser}(s, u)\}$

D. $\text{addclient} \stackrel{\text{def}}{=} \{(s, u) \mapsto s' \mid s' = (\pi_1(s), \pi_2(s) \cup \{(u, \emptyset)\}) \rightarrow \neg \text{hasUser}(s, u)\}$

E. $\text{addclient} \stackrel{\text{def}}{=} \{(s, u) \mapsto s' \mid s' = (\pi_1(s), \pi_2(s)) \wedge \neg \text{hasUser}(s, u)\}$

cu entao, em vez de user
 $\text{hasUser}(s, u)$, usar $\text{userExists}(u, s)$