

Cálculo de Sistemas de Informação

Perfil: MÉTODOS FORMAIS EM ENGENHARIA DE SOFTWARE

1.º/4.º Ano de MEI & MMC / MiEI, Universidade do Minho

Ano Lectivo de 2019/20

Exame de recurso — 30 de Janeiro

09h00

Sala E2-1.10

Importante — *Ler antes de iniciar a prova:*

- Esta prova consta de 8 questões que valem, cada uma, 2.5 valores. O tempo **médio** estimado para resolução de cada questão é de 15 min.
- Os alunos que desejem fazer melhoria apenas uma parte devem assinalá-lo na prova, que deverão entregar ao fim de uma hora.

PROVA COM CONSULTA (1 ou 2 horas)

Parte A

Questão 1 Considere a relação $R : X \rightarrow Y$ definida pela seguinte matriz Booleana

$$R = \begin{array}{c|ccccc} & A & B & C & D & E \\ \hline 1 & 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 1 & 0 & 0 \\ 3 & 0 & 1 & 0 & 1 & 0 \\ 4 & 0 & 0 & 1 & 0 & 0 \\ 5 & 0 & 0 & 0 & 0 & 0 \end{array}$$

onde $X = \{A, B, C, D, E\}$ e $Y = \{1..5\}$. Investigue a veracidade das seguintes afirmações:

1. R é uma função
2. R é difuncional
3. R é injectiva
4. R é sobrejectiva

Questão 2 Muitas funções binárias $A \times B \xrightarrow{f} C$ não são injectivas (por exemplo, a soma de inteiros) mas, se lhe fixarmos um dos argumentos (por exemplo o primeiro), são-no quanto ao outro argumento:

$$f(a, b) = f(a, b') \Rightarrow b = b'$$

Mostre que isso acontece se e só se

$$id \leq \langle \pi_1, f \rangle \tag{F1}$$

se verificar, onde $R \leq S$ compara relações quanto à sua injectividade.

Questão 3 Considere o operador relacional

$$\text{slice } S R = R \cap S / R^\circ \tag{F2}$$

definido em Alloy como se segue:

```
fun slice[s: A -> A, r: K -> A] : K -> A {
  { a : r.dom, b : a.r | (all b' : a.r | b' in s.b) }
}
```

É possível implementar *slice* em Haskell, com tipo

```
slice :: ((a, a) -> Bool) -> [(k, a)] -> [(k, a)]
```

onde a primeira relação é representada por um predicado binário. Calcule o teorema grátis de *slice*, instanciando-o para funções.

Questão 4 Mostre que a diferença de relações satisfaz a propriedade

$$(Q - R) - S = (Q - S) - R \tag{F3}$$

Parte B

Questão 5 Dada uma qualquer relação $B \xleftarrow{R} A$, a chamada *diagonal* $B \xleftarrow{R_d} A$ de R define-se por

$$R_d = R \cap (R \setminus R / R)^\circ \tag{F4}$$

e exibe a propriedade universal:

$$X \subseteq R_d \equiv \begin{cases} X \subseteq R \\ R \cdot X^\circ \cdot R \subseteq R \end{cases} \tag{F5}$$

Avalie a veracidade da afirmação seguinte:

A diagonal R_d é reflexiva se e só se R for uma pré-ordem.

Questão 6 Apresente justificações para a demonstração da propriedade

$$\delta \langle f \cdot R, \underline{k} \rangle = \delta R$$

que se segue:

$$\begin{aligned} & \delta \langle f \cdot R, \underline{k} \rangle \\ = & \{ \dots \} \\ & \ker (f \cdot R) \cap \ker \underline{k} \cap id \\ = & \{ \dots \} \\ & \delta (f \cdot R) \\ = & \{ \dots \} \\ & \delta (\delta f \cdot R) \\ = & \{ \dots \} \\ & \delta R \end{aligned}$$

□

Questão 7 O seguinte modelo em Alloy, que foi assunto das aulas da disciplina de Especificação e Modelação,

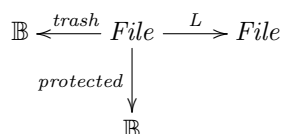
– File system trash can.

```

sig File {
  link : lone File
}
sig Trash extends File {}
sig Protected extends File {}

```

pode ser expresso pelo diagrama relacional,



onde L abrevia *link*: há ficheiros (*File*) que podem estar ligados entre si (L), apagados (*trash*) ou protegidos (*protected*). Converta para notação pointwise os *tipos relacionais* seguintes

$$\Phi_{\text{active}} \xleftarrow{L} \Phi_{\text{active}} \quad (\text{F6})$$

$$\Phi_{\text{active}} \xleftarrow{id} \Phi_{\text{protected}} \quad (\text{F7})$$

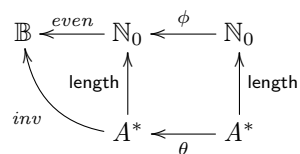
(onde $\text{active} = \neg \cdot \text{trash}$), e exprima por palavras suas, em cada caso, a propriedade do sistema que está a ser especificada.

Questão 8 Uma lista de pares $x \in (A \times A)^*$ pode ser representada simplesmente por $y \in A^*$ desde que o comprimento de y seja um número par (*even*). Seja $\theta :: A^* \rightarrow A^*$ a operação que remove o primeiro par de uma tal representação, quando esse par existe: $\theta = \text{drop } 2$.

Num dispositivo médico em que esta representação vai ser usada pretende-se garantir que θ preserva o invariante concreto:

$$\text{inv} = \text{even} \cdot \text{length} \quad (\text{F8})$$

Por *interpretação abstracta*, a prova pode ser feita encontrando uma *simulação* abstracta ϕ de θ no diagrama



que preserve *even*. Suponha que alguém conjecturou

$$\begin{aligned}
 \phi 0 &= 0 \\
 \phi 1 &= 0 \\
 \phi (n + 2) &= n
 \end{aligned}$$

(que claramente preserva *even*) e que, para demonstrar que ϕ simula θ , re-definiu ϕ de forma seguinte

$$\phi \cdot \text{in} = [0, \alpha] \quad (\text{F9})$$

$$\alpha \cdot \text{in} = [0, id] \quad (\text{F10})$$

onde $\text{in} = [0, \text{succ}]$.

Apresente justificações para os passos da prova apresentada de que ϕ simula θ :

$$\begin{aligned}
 & \text{length} \cdot \theta = \phi \cdot \text{length} \\
 \equiv & \quad \{ \dots \} \\
 & \text{length} \cdot \theta \cdot \text{length}^\circ \cdot \text{in} \subseteq [0, \alpha] \\
 \equiv & \quad \{ \dots \} \\
 & \begin{cases} \text{length}^\circ \cdot \underline{0} \subseteq (\text{length} \cdot \theta)^\circ \cdot \underline{0} \\ \text{length}^\circ \cdot \text{succ} \subseteq (\text{length} \cdot \theta)^\circ \cdot \alpha \end{cases} \\
 \equiv & \quad \{ \dots \} \\
 & \begin{cases} y = [] \Rightarrow \theta y = [] \\ \text{length}^\circ \cdot \text{succ} \cdot \text{in} \subseteq (\text{length} \cdot \theta)^\circ \cdot [0, id] \end{cases} \\
 \equiv & \quad \{ \dots \} \\
 & \begin{cases} \text{drop 2 } [] = [] \\ \begin{cases} \text{length}^\circ \cdot \underline{1} \subseteq (\text{length} \cdot \theta)^\circ \cdot \underline{0} \\ \text{length}^\circ \cdot (2+) \subseteq (\text{length} \cdot \theta)^\circ \end{cases} \end{cases} \\
 \equiv & \quad \{ \dots \} \\
 & \begin{cases} \text{length } y = 1 \Rightarrow \text{length } (\theta y) = 0 \\ \text{length}^\circ \cdot (2+) \subseteq (\text{length} \cdot \theta)^\circ \end{cases} \\
 \equiv & \quad \{ \dots \} \\
 & \begin{cases} \text{drop 2 } [-] = [] \\ \text{length } y = 2 + n \Rightarrow \text{length } (\text{drop 2 } y) = n \end{cases} \\
 \equiv & \quad \{ \dots \} \\
 & \text{true}
 \end{aligned}$$

NB: assumas propriedades básicas da função drop.