Métodos Formais de Programação II

4.º Ano da LMCC (7008N2)
Ano Lectivo de 1998/99

Exame (época normal, 2.ª chamada) — 1 de Julho de 1999
09h30

NB: *Esta prova consta de* **5** *questões de* **4 valores** *cada uma.*
Sempre que os seus cálculos utilizarem resultados, leis, teoremas etc. *disponíveis no material pedagógico desta disciplina não os copie para a sua folha de exame — limite-se a citar a sua referência.*

PROVA COM CONSULTA ($3\ horas$)

Questão 1 Na teoria de refinamento estudada nesta disciplina a comparação entre dois tipos de dados A e B é feita com base na existência de uma *função de abstracção* f sobrejectiva, isto é, invertível à direita:

$$A \unlhd B \quad sse \quad \exists\ A \xleftarrow{f} B\ ,\ A \xrightarrow{r} B\ :\ f \cdot r = id_A$$

Sejam f e g duas funções de abstracção (sobrejectivas), F um functor e p um predicado. Indique, justificando formalmente, quais das expressões funcionais seguintes designam também funções de abstracção (sobrejectivas):

$$1 + (\mathsf{F}\, f) \tag{1}$$
$$f \cdot g \tag{2}$$
$$p \rightarrow f, g \tag{3}$$

Questão 2 Seja $\mathsf{Bt}\ A$ o tipo de dados *árvore binária de As* que foi assunto da Questão 1 do exame de 1.ª-chamada e se pode definir pelo isomorfismo

$$\mathsf{Bt}\ A \xleftarrow{[empty, node]} 1 + A \times (\mathsf{Bt}\ A \times \mathsf{Bt}\ A)$$

Uma dada árvore binária $t \in \mathsf{Bt}\ A$ dir-se-á uma *árvore binária de procura* se estiver equilibrada ($balanced$) e ordenada ($ordered$):

$$abp\, t \stackrel{\rm def}{=} (balanced\, t) \wedge (ordered\, t)$$

isto é, $abp \stackrel{\rm def}{=} \wedge \cdot \langle balanced, ordered \rangle$.

Estar equilibrada significa que em qualquer dos seus nós o *peso* das duas sub-árvores (esquerda e direita) difere quando muito de uma unidade. Estar ordenada significa que o valor que se encontra em cada um dos seus nós é estritamente maior do que qualquer valor da sua sub-árvore esquerda e estritamente menor do que qualquer valor da sua sub-árvore direita.

1. Especifique o predicado $ordered$ em notação livre de variáveis.
2. É fácil verificar que o teste do invariante abp é muito ineficiente. Use as leis de Fokkinga e "banana-split" para calcular uma implementação de abp que faça uma única visita à árvore argumento.

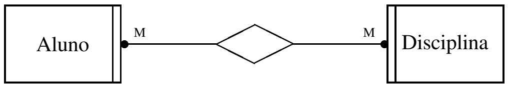
Questão 3 Sabendo que $\otimes$ é a função $s \otimes t \stackrel{\rm def}{=} \{(a,b) \mid\ a \in s \wedge b \in t\}$, complete as reticências do diagrama funcional que se segue

$$\begin{tikzcd} \ldots \arrow[r, "collect \times collect"'] \arrow[rrrrr, bend left, "f"] & \ldots \arrow[r, "\bowtie"'] & \ldots \arrow[r, "A \rightharpoonup \otimes"'] & \ldots \arrow[r, "discollect"'] & \ldots \arrow[r, "2^{flatr}"'] & \ldots \end{tikzcd}$$

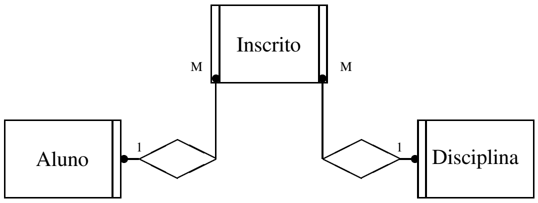
e explique, através de um exemplo, qual o significado da função f.

Será f um epimorfismo? E um isomorfismo? Justifique.

Questão 4 Considere os seguintes diagramas E-R:



e



Conterão estes diagramas exactamente a "mesma" informação? Compare-os formalmente através da sua conversão para notação SETS.

Questão 5 Uma fábrica de cabos condutores eléctricos pretende uma aplicação para controlo dos seus 'stocks'. Há vários tipos de cabos condutores, cada um identificado por um *código de artigo*. Cada segmento de cabo constitui uma *ponta*, é de um determinado tipo, tem um dado comprimento (múltiplo de $1m$) e está armazenado algures numa *bobine*. Cada *bobine* tem uma *matrícula* que a identifica e está num determinado *estado*, a saber: armazenada num dado *armazém*, em manutenção, em produção, em controlo de qualidade, no cliente, abatida ou morta. Na mesma bobine só podem estar armazenadas pontas do mesmo tipo. Todas as bobines do mesmo tipo estão armazenadas no mesmo armazém.

1. Complete a especificação seguinte da estrutura de base para gestão deste 'stock',

$$\begin{array}{rcl} DBase & \cong & \#Art \rightharpoonup \#Arm \times Bobines \\ Bobines & \cong & \#Bob \rightharpoonup Ponta^{\star} \times Estado \\ Ponta & \cong & \ldots \\ Estado & \cong & \ldots \end{array}$$

   onde *#Art (código de artigo)*, *#Arm (referência de armazém)* e *#Bob (matrícula de bobine)* são tipos primitivos.

2. Escreva os dois invariantes que estão implícitos nas duas últimas frases do texto.
3. Aplique o cálculo estudado nesta disciplina à derivação formal de uma implementação relacional de $DBase$.
