

**Métodos Formais de Programação I +  
Opção I - Métodos Formais de Programação I**

4.º Ano da LMCC (7007N2) + LES1 (5307P6)  
Ano Lectivo de 2001/02

Exame (2.ª chamada) — 30 de Janeiro de 2002  
14h30  
Salas 2110 e 2111

**NB:** Esta prova consta de 7 alíneas todas com o mesmo valor.

PROVA SEM CONSULTA (2 horas)

**Questão 1** Recorde o *Problema 5* das sessões laboratoriais desta disciplina, onde se especificou um sistema bancário “brinquedo”:

```
BAMS = map AccId to Account;  
Account :: H: set of AccHolder  
         B: Amount;  
  
AccId    = seq of char;  
AccHolder = seq of char;  
Amount = int;
```

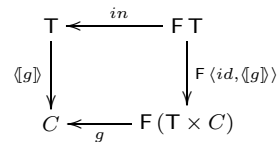
1. No sentido de tornar o modelo mais realista, acrescentou-se-lhe informação sobre clientes:

```
BAMS' :: accounts: map AccId to Account  
       accholders: map AccHolder to AccHolderInfo;  
AccHolderInfo :: name: seq of char  
               passport: seq of char  
               address: seq of char;
```

Falta, contudo, um invariante que garanta a integridade referencial entre *accounts* e *accholders*. Especifique esse invariante sobre *BAMS'*.

2. Na sequência da alínea anterior, melhore ainda mais o modelo substituindo *B: Amount* por um registo de movimentos de crédito/débito, devidamente datados, e especifique, para esse modelo, a operação que devolve o saldo actual de uma dada conta.

**Questão 2** Recorde o conceito geral de *paramorfismo* de “gene” *g*:



1. Especifique em VDM-SL a função

`inseg : nat -> seq of nat`

(“inseg” = ‘initial segment’) que gera a lista decrescente de todos os números naturais iguais ou inferiores a um dado número, e caracterize-a como um paramorfismo (para  $T = \mathbb{N}$ ).

2. Represente através de um diagrama o paramorfismo *nw* que calcula o número de palavras de um texto:

```

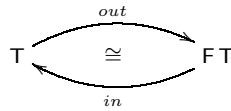
nw : seq of char -> nat
nw(s) == if s = [] then 0
         else if not sep(hd s) and sepahead(tl s)
              then nw(tl s) + 1 else nw(tl s) ;

sepahead: seq of char -> bool
sepahead(s) == (s = []) or sep(hd s) ;

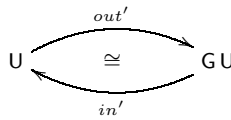
sep : char -> bool
sep(c) == c = ' ' or c = '\n' or c = '\t' ;

```

3. É um resultado conhecido que todo o paramorfismo sobre um tipo



é convertível num hilomorfismo de tipo



onde G se obtém de F da forma seguinte:

$$\begin{cases} GX = F(T \times X) \\ Gf = F(id \times f) \end{cases}$$

Complete as reticências no seguinte processo de transformação de  $\langle g \rangle$  num tal hilomorfismo:

$$\begin{aligned} & \langle g \rangle \cdot in = g \cdot F \langle id, \langle g \rangle \rangle \\ = & \{ \dots \} \\ & \langle g \rangle = g \cdot F \langle id, \langle g \rangle \rangle \cdot out \\ = & \{ \dots \} \\ & \langle g \rangle = g \cdot F ((id \times \langle g \rangle) \cdot \langle id, id \rangle) \cdot out \\ = & \{ \dots \} \\ & \langle g \rangle = g \cdot F (id \times \langle g \rangle) \cdot F \langle id, id \rangle \cdot out \\ = & \{ \dots \} \\ & \langle g \rangle = g \cdot (G \langle g \rangle) \cdot F \langle id, id \rangle \cdot out \\ = & \{ \dots \} \\ & \langle g \rangle = [ g, F \langle id, id \rangle \cdot out ] \\ = & \{ \dots \} \\ & \langle g \rangle = \langle g \rangle \cdot [ F \langle id, id \rangle \cdot out ] \end{aligned}$$

**Questão 3** O registo de chamadas não atendidas em telemóveis é vulgarmente organizado de forma a que os últimos números que chamaram estejam imediatamente acessíveis. Essa estrutura corresponde à de um “stack” cuja operação de “push” se redefine da forma seguinte:

```

npush : nat * seq of nat -> seq of nat
-- push nat or move it to the front
npush(n, s) == [n] ^ [ s(i) | i in set inds s & n <> s(i) ] ;

```

1. Descreva por palavras suas a propriedade da função  $npush$  que é registada pela igualdade

$$npush \cdot \langle \pi_1, npush \rangle = npush$$

e demonstre-a.

2. Considere agora a estrutura

```
TLM :: registo: set of Call
      agenda: map nat to seq of char ;
Call :: number: nat
      time: nat ;
```

que especifica o registo das chamadas não atendidas e respectiva hora de chamada, assim como a agenda que o telemóvel tem indicando, para alguns números de telefone, os nomes respectivos. Especifique em VDM-SL a operação de “display no écran” das chamadas não atendidas:

```
display : TLM -> seq of seq of char
```

**NB:** Como é previsível, só vai poder mostrar um nome se o respectivo número estiver na agenda. Recorra a funções auxiliares genéricas, se necessário.

## Anexo—Cálculo de Funções

### COMPOSIÇÃO

$$\text{Natural-id} \quad f \cdot id = id \cdot f = f \quad (1)$$

$$\text{Associatividade} \quad (f \cdot g) \cdot h = f \cdot (g \cdot h) \quad (2)$$

### PRODUTO

$$\text{Universal-}\times \quad k = \langle f, g \rangle \Leftrightarrow \begin{cases} \pi_1 \cdot k = f \\ \pi_2 \cdot k = g \end{cases} \quad (3)$$

$$\text{Cancelamento-}\times \quad \pi_1 \cdot \langle f, g \rangle = f \quad , \quad \pi_2 \cdot \langle f, g \rangle = g \quad (4)$$

$$\text{Reflexão-}\times \quad \langle \pi_1, \pi_2 \rangle = id_{A \times B} \quad (5)$$

$$\text{Fusão-}\times \quad \langle g, h \rangle \cdot f = \langle g \cdot f, h \cdot f \rangle \quad (6)$$

$$\text{Absorção-}\times \quad (i \times j) \cdot \langle g, h \rangle = \langle i \cdot g, j \cdot h \rangle \quad (7)$$

$$\text{Functor-}\times \quad (g \cdot h) \times (i \cdot j) = (g \times i) \cdot (h \times j) \quad (8)$$

$$\text{Functor-id-}\times \quad id_A \times id_B = id_{A \times B} \quad (9)$$

### COPRODUTO

$$\text{Universal-}\text{+} \quad k = [f, g] \Leftrightarrow \begin{cases} k \cdot i_1 = f \\ k \cdot i_2 = g \end{cases} \quad (10)$$

$$\text{Cancelamento-}\text{+} \quad [g, h] \cdot i_1 = g \quad , \quad [g, h] \cdot i_2 = h \quad (11)$$

$$\text{Reflexão-}\text{+} \quad [i_1, i_2] = id_{A+B} \quad (12)$$

$$\text{Fusão-}\text{+} \quad f \cdot [g, h] = [f \cdot g, f \cdot h] \quad (13)$$

$$\text{Absorção-}\text{+} \quad [g, h] \cdot (i + j) = [g \cdot i, h \cdot j] \quad (14)$$

$$\text{Functor-}\text{+} \quad (g \cdot h) + (i \cdot j) = (g + i) \cdot (h + j) \quad (15)$$

$$\text{Functor-id-}\text{+} \quad id_A + id_B = id_{A+B} \quad (16)$$

### EXPONENCIAÇÃO

$$\text{Universal} \quad k = \bar{f} \Leftrightarrow f = ap \cdot (k \times id) \quad (17)$$

$$\text{Cancelamento} \quad f = ap \cdot (\bar{f} \times id) \quad (18)$$

<b>Reflexão</b>	$\overline{ap} = id_{BA}$	(19)
<b>Fusão</b>	$\overline{g \cdot (f \times id)} = \overline{g} \cdot f$	(20)
<b>Absorção</b>	$f^A \cdot \overline{g} = \overline{f \cdot g}$	(21)
<b>Functor</b>	$(g \cdot h)^A = g^A \cdot h^A$	(22)
<b>Functor-id</b>	$id^A = id$	(23)

INDUÇÃO

<b>Universal-cata</b>	$k = \langle \alpha \rangle \Leftrightarrow k \cdot in = \alpha \cdot F k$	(24)
<b>Cancelamento-cata</b>	$\langle \alpha \rangle \cdot in = \alpha \cdot F \langle \alpha \rangle$	(25)
<b>Reflexão-cata</b>	$\langle in \rangle = id_{\mu F}$	(26)
<b>Fusão-cata</b>	$f \cdot \langle \alpha \rangle = \langle \beta \rangle \quad \text{if} \quad f \cdot \alpha = \beta \cdot F f$	(27)