Métodos Formais de Programação II +
Opção - Métodos Formais de Programação II

4.º Ano de LMCC (7008N2) + LESI (5308P3)
Ano Lectivo de 2007/08

Exame (época de recurso) — 15 de Julho 2008
09H30
Sala 2210

NB: *Esta prova consta de **8** alíneas todas com a mesma cotação.*

PROVA SEM CONSULTA (2 horas)

Questão 1 Diz-se que uma especificação S é implementada por R, escrevendo-se $S \vdash R$, sempre que R é mais definida e mais determinística que S, onde esta está definida. Formalmente:

$$S \vdash R \quad \equiv \quad (\delta\, S \subseteq \delta\, R) \,\wedge\, (R \cdot \delta\, S \subseteq S) \tag{1}$$

Seja S a relação especificada pelo seguinte par **pre**/**post**:

$$\begin{array}{l} S : (y : I\!\!R) \leftarrow (x : I\!\!R) \\ \textbf{pre}\ \textsc{True} \\ \textbf{post}\ \ y - x > 2 \end{array}$$

Calcule a condição mais geral a que devem obedecer os parâmetros a e b da função

$$\begin{array}{l} f : I\!\!R \leftarrow I\!\!R \\ f\ x \stackrel{\mathrm{def}}{=} a \times x + b \end{array}$$

por forma a que $S \vdash f$ se verifique.

Questão 2 É conhecido o papel da lei

$$\begin{tikzcd} A \rightharpoonup (D \times (B \rightharpoonup C)) \arrow[r, bend left, "\triangle_n"] \arrow[r, phantom, "\leq"] & (A \rightharpoonup D) \times (A \times B \rightharpoonup C) \arrow[l, bend left, "\bowtie_n"] \end{tikzcd} \tag{2}$$

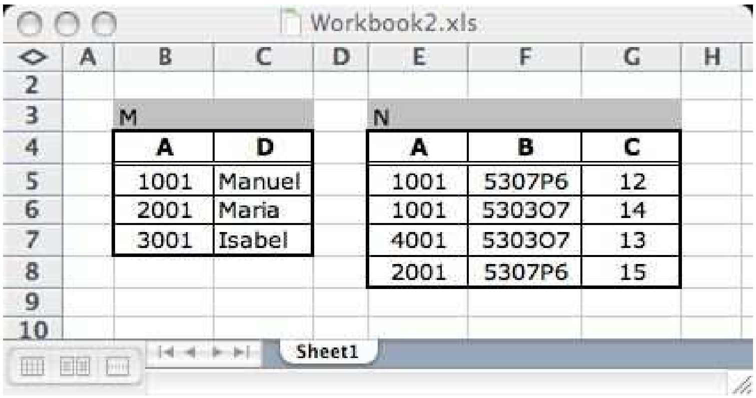
na decomposição de modelos de dados com vista à sua implementação em RDBMS.

1. Escrita em notação VDM, (2) fica

$$\begin{tikzcd} \mathtt{map}\ A\ \mathtt{to}\,(D * (\mathtt{map}\ B\ \mathtt{to}\, C)) \arrow[r, bend left, "unnjoin"] \arrow[r, phantom, "\leq"] & (\mathtt{map}\ A\ \mathtt{to}\, D) * (\mathtt{map}\ A * B\ \mathtt{to}\, C) \arrow[l, bend left, "njoin"] \end{tikzcd} \tag{3}$$

Defina, em VDM, uma das funções `njoin` e `unnjoin`, à sua escolha.

2. É fácil ver que o par (M, N) que abaixo se ilustra,



M

A	D
1001	Manuel
2001	Maria
3001	Isabel

N

A	B	C
1001	5307P6	12
1001	5303O7	14
4001	5303O7	13
2001	5307P6	15

nunca poderia ter sido gerado por $unnjoin$. Diga porquê e identifique um facto que conhece sobre $unnjoin$ que este contra-exemplo contesta.

3. Para que $njoin$ realize um *join* sem perdas — isto é, tal que $unnjoin(njoin(M, N)) = (M, N)$ se verifique — é preciso que esteja garantido o invariante descrito pelo rectângulo superior do diagrama que se segue:

$$\begin{tikzcd} & A & A \times B \arrow[l, "\pi_1"'] & \\ D & A \arrow[l, "M"] \arrow[u, "\delta\, M"] \arrow[ur, phantom, "\supseteq"] & A \times B \arrow[l, "\pi_1"] \arrow[u, "\delta\, N"'] \arrow[r, "N"'] & C \end{tikzcd}$$

isto é, $\pi_1 \cdot \delta\, N \subseteq \delta\, M \cdot \pi_1$

Complete os cálculos seguintes que convertem esta versão do invariante em notação VDM:

$\pi_1 \cdot \delta\, N \subseteq \delta\, M \cdot \pi_1$

$\equiv$ { .... }

$\delta\, N \subseteq \pi_1^\circ \cdot \delta\, M \cdot \pi_1$

$\equiv$ { .... }

$\langle \forall\, x, y \;::\; y(\delta\, N)x \Rightarrow y(\pi_1^\circ \cdot \delta\, M \cdot \pi_1)x \rangle$

$\equiv$ { .... }

.... (complete com vários passos que faltam) ....

$\equiv$ { .... }

```
forall a,b & mk_(a,b) in set dom N => a in set dom M
```

Questão 3 Um algoritmo conhecido para compressão de sequências ordenadas redu-las a conjuntos de intervalos, omitindo todos os valores entre os extremos de uma série de valores consecutivos. Exemplificando, a sequência

$$[4, 5, 6, 10, 20, 21, 22, 23, 24]$$

será reduzida a

$$\{(4, 6), (10, 10), (20, 24)\}$$

1. Complete a seguinte especificação em VDM-SL de uma função que deverá realizar tal compressão, onde o tipo de dados `int` é uma simplificação de um qualquer outro tipo paramétrico totalmente ordenado:

```
compress : seq of int -> set of (int*int)
compress(l) == ... g(...) ... ;

g : int * set of (int*int) -> set of (int*int)
g(a,s)  == let r = { p | p in set s & p.#2 = a-1 }
              in ....
```

2. Investigue o comportamento da sua versão de `compress` quando a lista de entrada **não** está ordenada. Poderá `compress` ser considerada uma **função** de abstracção? E de representação?

Questão 4 Considere o seguinte modelo formal que especifica, em notação VDM-SL, (a versão simplificada de) um sistema para gestão de congressos, conferências, simpósios, *etc.*:

```
Plan  = map Date to Day ;          /* cada dia tem o seu programa     */
Day   = map Time to Inf ;          /* programa hora a hora            */
Inf = InvSpeaker | Paper | Other ;
Paper :: A: seq of Author          /* lista de autores                */
         T: Title                  /* título do artigo                */
         S: Abstract ;             /* sumário                         */
InvSpeaker :: A: Author            /* orador convidado                */
              T: Title             /* título da sua apresentação      */
              S: Abstract ;        /* sumário                         */
Other :: O: token ;                /* S indica intervalo de café, etc */
Time  = int ;
Date  = seq of char ;
Author  = seq of char ;
Title  = seq of char ;
Abstract  = seq of char ;
```

Sabe-se que está garantido, por invariante, que cada dia tem programa não vazio.

Calcule, usando as leis de refinamento estudadas nesta disciplina, uma implementação relacional de `Plan`. Apresente, para cada passo, indicação das leis que usou e/ou as respectivas funções de abstracção e de representação.

Questão 5 Considere o seguinte modelo VDM que especifica, abstractamente, a estrutura de um sistema de informação que, baseado no 'World Wide Web', dá a garantia de **integridade referencial**:

```
WWW  = map Ref to URL           -- URL=Universal Resource Location
       inv M == forall k in set dom M &
                    (exists i in set inds M(k) &
                        is_HyperLink(M(k)(i))) => M(k)(i).link in set dom M;
URL  = seq of Unit;
Unit = PlainText | HyperLink;
PlainText = seq of Word;
Word = seq of char;
HyperLink :: link: Ref           -- reference to another site
             txt: PlainText;     -- "underlined text"
Ref = token ;
```

Sabendo que

$$x \in l \quad \equiv \quad \langle \exists\, i \,:\, i \in \; inds\, l : \; x \in (l\; i) \rangle \tag{4}$$

é a relação de pertença associada a listas, complete o cálculo que se segue e que mostra que o invariante sobre o tipo `WWW` é a conversão para VDM *pointwise* de

$$inv\ M \stackrel{\mathrm{def}}{=} (\in \cdot M)^\circ \preceq M$$

$$\begin{tikzcd}
Ref \arrow[r, "M"] \arrow[dr, "\in \cdot M"'] & (Word^\star + Ref \times Word^\star)^\star \arrow[d, "\in"] \\
& Ref
\end{tikzcd}$$

$$\begin{array}{ll}
\equiv & \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\
& \delta\,((\in \cdot M)^\circ) \subseteq \delta\, M \\
\equiv & \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\
& (\in \cdot M)^\circ \subseteq \top \cdot M \\
\equiv & \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\
& k'(\in \cdot M)k \ \Rightarrow\ k(\top \cdot M)k' \\
\equiv & \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\
& \langle \exists\, x \ :: \ k' \in x \ \wedge\ x\ M\ k \rangle \ \Rightarrow\ k' \in dom\ M \\
\equiv & \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\
& (k \in dom\ M \ \wedge\ k' \in M\ k) \ \Rightarrow\ k' \in dom\ M \\
\equiv & \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\
& (k \in dom\ M \ \wedge\ \langle \exists\, i \ :\ i \in inds(Mk) :\ k' \in (M\ k)i \rangle) \ \Rightarrow\ k' \in dom\ M \\
\equiv & \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\
& (k \in dom\ M \ \wedge\ \langle \exists\, i \ :\ i \in inds(Mk) :\ k'[\bot\ , \in]\ (M\ k)i \rangle) \ \Rightarrow\ k' \in dom\ M \\
\equiv & \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\
& (k \in dom\ M \ \wedge\ \langle \exists\, i \ :\ i \in inds(Mk) :\ k'\ (\in \cdot i_2^\circ)\ (M\ k)i \rangle) \ \Rightarrow\ k' \in dom\ M \\
\equiv & \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\
& (k \in dom\ M \ \wedge\ \langle \exists\, i \ :\ i \in inds(Mk) :\ \langle \exists\, y \ ::\ (M\ k)i = i_2\ y \ \wedge\ k' = \pi_1\ y \rangle\rangle) \ \Rightarrow\ k' \in dom\ M \\
\equiv & \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\
& (k \in dom\ M \ \wedge\ \langle \exists\, i \ :\ i \in inds(Mk) :\ \langle \exists\, y \ ::\ (M\ k)i = i_2\ y \rangle\rangle) \ \Rightarrow\ (\pi_1\ y) \in dom\ M
\end{array}$$
