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

4.º Ano da LMCC (7007N2) + LESI (5307P6)
Ano Lectivo de 2006/07

Exame (época de recurso) — 16 de Fevereiro 2007
09h30
Sala 1315

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

PROVA SEM CONSULTA (2 horas)

Questão 1 Dada o par **pre/post** seguinte, escrito em VDM-SL

```
MaxNatSet(s: set of nat) m: nat
pre s <> {}
post m in set s and forall a in set s & a <= m ;
```

1. Mostre que a transformada-PF do predicado **post** é a expressão

$$\in \cap (\in \setminus \leq)^{\circ} \tag{1}$$

onde $\in$ designa a pertença de conjuntos (o `... in set ...` do VDM-SL) e $R \setminus S$ é a operação de divisão de relações que estudou nas aulas práticas desta disciplina

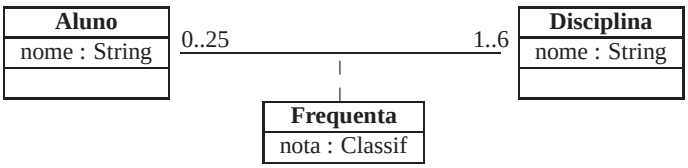
2. Demonstre, por igualdade indirecta e uso extensivo das conecções de Galois que conhece, a propriedade

$$(R \cup S) \setminus T \quad = \quad (R \setminus T) \cap (S \setminus T) \tag{2}$$

isto é, construa os vários passos que medeiam

$$\begin{array}{rl} & X \subseteq (R \cup S) \setminus T \\ \equiv & \vdots \\ & X \subseteq (R \setminus T) \cap (S \setminus T) \end{array}$$

Questão 2 Numa questão de um exame anterior desta disciplina pedia-se para se especificar um modelo de dados em notação VDM-SL que captasse exactamente o significado do diagrama UML



As respostas a esta questão incluiram, com maior or menor qualidade, os tipos de dados

```
Classif = <F> |<R> | <D> | Positiva ;
Positiva = nat
        inv n == n in set {10,...,20};
```

mas diferiram quanto à modelação do resto do diagrama, sendo as alteratívas mais populares as seguintes:

- Alternativa A:

```
StateA :: alunos:      map AId to Nome
          disciplinas: map DId to Nome
          frequenta:   Classifs
       inv mk_StateA(M,N,R) ==
              {d | mk_(-,d) in set dom R} subset dom N  and
              {a | mk_(a,-) in set dom R} subset dom M;

Classifs = map (AId * DId) to Classif
       inv R == forall mk_(a,d) in set dom R &
                   let x = card {d' | mk_(a',d') in set dom R & a = a'},
                       y = card {a' | mk_(a',d') in set dom R & d = d'}
                   in  y <= 25 and x >= 1 and x <= 6;
```

- Alternativa B:

```
StateB :: alunos:      map AId to Aluno
          disciplinas: map DId to Nome
       inv mk_StateB(M,N) ==
              let D = dunion { dom M(a).frequenta | a in set dom M }
              in  D subset dom N and
                  (forall d in set D &
                          card { a | a in set dom M &
                                 d in set dom M(a).frequenta } <= 25)
                  and
                  forall a in set dom M &
                       let D = dom M(a).frequenta
                       in  card D >= 1 and card D <= 6;

Aluno :: nome: Nome
         frequenta: map DId to Classif;
```

para

```
AId = token;
DId = token;
Nome =token;
```

1. Pode mostrar-se que `StateA` e `StateB` são isomorfos. Como parte dessa prova, complete a definição da função

   ```
   repA2B : StateA -> StateB
   repA2B(mk_StateA(M,N,R)) == is not yet implemented;
   ```

   que representa dados do modelo A no modelo B.

2. Especifique, sobre cada um dos modelos A e B, a operação que cancela a inscrição do aluno `a` na disciplina `d` que `a` está a frequentar.

Questão 3 Considere o operador de "actualização selectiva" de uma função finita, em notação VDM-SL:

```
selUp[@A,@B]: set of @A * (@B -> @B) * map @A to @B -> map @A to @B
selUp(S,f,M) == M ++ { k |-> f(M(k)) | k in set dom M inter S}
```

Partindo da semântica relacional deste operador,

$$\lceil selUp(S,f,M)\rceil \quad = \quad M \dagger (f \cdot M \cdot \lceil S\rceil) \tag{3}$$

complete o cálculo que se segue de uma condição necessária e suficiente, em VDM-SL, para que

$$selUp(S,f,M) = M \tag{4}$$

se verifique:

$M \dagger (f \cdot M \cdot \lceil S\rceil) = M$

$\equiv$ { ..... }

$f \cdot M \cdot \lceil S\rceil \subseteq M$

$\equiv$ { ..... }

$f \cdot M \cdot \lceil S\rceil \cdot \delta\, M \subseteq M$

$\equiv$ { ..... }

$f \cdot M \cdot \lceil S\rceil \cdot M^\circ \subseteq id$

$\equiv$ { ..... }

$f \cdot \mathsf{img}\,(M \cdot \lceil S\rceil) \subseteq id$

$\equiv$ { ..... }

$\rho\,(M \cdot \lceil S\rceil) \subseteq f^\circ$

$\equiv$ { ..... }

```
forall b in set rng (S <:  M) & f(b) = b
```

Questão 4 Demonstre o resultado seguinte:

$$R \cup S \text{ é injectiva} \quad \equiv \quad R \text{ é injectiva } \wedge\ S \text{ é injectiva } \wedge\ R^\circ \cdot S \subseteq id \tag{5}$$

Questão 5 Como sabe, o VDM-SL permite manipular listas por indexação dos seus elementos, cf. o seguinte quadro adaptado do respectivo manual *on-line*:

Operator	Name	Semantics description
`inds l`	Indexes	yields the set of indexes of `l`, i.e. the set `{1,...,len l}`.
`l(i)`	Sequence application	yields the element of index from `l`. `i` must be in the indexes of `l`
`l ++ m`	Sequence modification	the elements of `l` whose indexes are in the domain of `m` are modified to the range value that the index maps into. `dom m` must be a subset of `inds l`

1. Apresente a sua própria especificação (não recursiva) do operador de *sequence modification* completando

```
smod[@A]: seq of @A * map nat1 to @A -> seq of @A
smod(l,m) == ....
pre ... ;
```

2. A indexação permite ver sequências como relações simples (finitas) entre posições e elementos a sequenciar; por exemplo, a sequência $[a, b, a]$ é vista como a relação $\{(a, 1), (b, 2), (a, 3)\}$. Logo, operações sobre sequências podem ser expressas como operações sobre relações simples, por exemplo $cons(a, l) = [a] \frown l$ expressa por

$$cons(a, L) = \underline{a} \cdot \underline{1}^{\circ} \cup L \cdot succ^{\circ} \tag{6}$$

onde a relação L representa a sequência l e $succ = (1+)$; mais ainda, invariantes sobre sequências podem exprimir-se como propriedades de relações simples, por exemplo:

$$l \text{ não tem elementos repetidos} \quad \equiv \quad L \text{ é injectiva} \tag{7}$$

Complete o seguinte cálculo da pré-condição a exigir a $cons$ para garantir a manutenção do invariante (7) acima:

$cons(a, l)$ não tem elementos repetidos

$\equiv$ { ..... }

$\underline{a} \cdot \underline{1}^{\circ} \cup L \cdot succ^{\circ}$ é injectiva

$\equiv$ { ..... }

$\underline{a} \cdot \underline{1}^{\circ}$ é injectiva $\wedge$ $L \cdot succ^{\circ}$ é injectiva $\wedge$ $(\underline{a} \cdot \underline{1}^{\circ})^{\circ} \cdot L \cdot succ^{\circ} \subseteq id$

$\equiv$ { ..... }

$\underline{1} \cdot \underline{a}^{\circ} \cdot \underline{a} \cdot \underline{1}^{\circ} \subseteq id \ \wedge \ suc \cdot L^{\circ} \cdot L \cdot succ^{\circ} \subseteq id \ \wedge \ \underline{a}^{\circ} \cdot L \subseteq \underline{1}^{\circ} \cdot succ$

$\equiv$ { ..... }

$\underline{a}^{\circ} \cdot \underline{a} \subseteq \underline{1}^{\circ} \cdot \underline{1} \ \wedge \ L^{\circ} \cdot L \subseteq succ^{\circ} \cdot succ \ \wedge \ \underline{a}^{\circ} \cdot L \subseteq \underline{1}^{\circ} \cdot succ$

$\equiv$ { ..... }

TRUE $\wedge$ $L^{\circ} \cdot L \subseteq id \ \wedge \ \underline{a}^{\circ} \cdot L \subseteq \underline{1}^{\circ} \cdot succ$

$\equiv$ { L é injectiva por hipótese (invariante à entrada) }

$\underline{a}^{\circ} \cdot L \subseteq \underline{1}^{\circ} \cdot succ$

$\equiv$ { ..... }

$\langle \forall \, n \ : \ n \in \delta \, L : \ a \, L \, n \Rightarrow 1 = 1 + n \rangle$

$\equiv$ { ..... }

$\langle \forall \, n \ : \ n \in \mathsf{inds} \, l : \ l(n) = a \Rightarrow n = 0 \rangle$

$\equiv$ { ..... }

$\langle \forall \, n \ : \ n \in \mathsf{inds} \, l : \ \neg(l(n) = a) \rangle$