Métodos Formais de Programação II +
Opção II — Métodos Formais de Programação II +
Especificação e Semântica

4.º Ano da LMCC (7008N2) + LESI (5308P3) + LMCC (7007A4)
Ano Lectivo de 1996/97

Exame (época de recurso) — 12 de Setembro de 1997
09h30
Sala 2205

Importante:
Os alunos que pretendem ser avaliados à disciplina Métodos Formais de Programação II *(7008N2) ou* Opção II — Métodos Formais de Programação II *(5308P3) devem resolver as questões 2, 7, 8, 9, 10, 11 e 12.*

Os alunos que pretendem ser avaliados à disciplina Especificação e Semântica *(7007A4) devem resolver as questões 1, 2, 3, 4, 5, 6 e 7.*

PROVA COM CONSULTA (3 $horas$)

Questão 1 Mostre, através de um contra-exemplo, que o isomorfismo

$$(A^\star)^2 \quad \cong \quad 2^\star \times A^\star$$

é, em geral, falso.

Questão 2 Mostre que, em SETS, a resolução em ordem a X do sistema de equações

$$\begin{cases} X & \cong & Y \times B \\ Y & \cong & 1 + A \times Y \end{cases}$$

conduz a

$$X \quad \cong \quad B + A \times X$$

para quaisquer A e B.

Questão 3 Recorde a função seguinte que extrai de uma sequência l a sua subsequência de n elementos (ou os que fôr possível extrair) que começa na i-ésima posição:

$$subl : A^\star \times \mathbb{N} \times \mathbb{N}_0 \to A^\star$$
$$subl(l,i,n) \stackrel{\text{def}}{=}$$
$$\begin{cases} n = 0 \lor l = <> & \Rightarrow & <> \\ n > 0 \land l \neq <> & \Rightarrow & let & h = head(l) \\ & & & t = tail(l) \\ & & in & \begin{cases} i = 1 & \Rightarrow & <h> \frown subl(t,i,n-1) \\ i > 1 & \Rightarrow & subl(t,i-1,n) \end{cases} \end{cases}$$

Mostre (*e.g.* por indução em $A^\star$) que o seguinte facto se verifica:

$$subl(l,i,n+m) \quad = \quad subl(l,i,n) \frown subl(l,i+n,m) \tag{1}$$

Questão 4 Considere a seguinte especificação de uma função em SETS baseada em $subl$:

$$split(l,n) \stackrel{\text{def}}{=} \begin{array}{ll} let & c = length(l) \\ & i = div(c,n) \\ in & <subl(l,(j-1)\times n+1,n) \mid j \leftarrow inseq(\left\{ \begin{array}{lcl} rem(c,n)=0 & \Rightarrow & i \\ \neg(rem(c,n)=0) & \Rightarrow & i+1 \end{array} \right.)> \end{array}$$

onde div e rem são as conhecidas funções de aritmética inteira e $inseq$ é a função

$$inseq(i) \stackrel{\text{def}}{=} \left\{ \begin{array}{lcl} i=0 & \Rightarrow & <> \\ \neg(i=0) & \Rightarrow & inseq(n-1) \frown <i> \end{array} \right.$$

1. Após calcular $split(<2,5,4>,2)$, tipifique $split$, *i.é* preencha as reticências em

$$split \quad : \quad \cdots \longrightarrow \cdots$$

2. Explique por palavras suas o significado das funções $inseq$ e $split$.
3. Mostre (*e.g.* por apresentação de um contra-exemplo) que o seguinte facto não se verifica:, em geral:

$$elems(length^\star(split(l,n))) \quad = \quad \{n\}$$

Questão 5 Considere o seguinte fragmento de um modelo $GIC : IGIC \to Sets$ para definição e manipulação de *gramáticas independentes de contexto*, onde SYM designa em $Sets$ um domínio primitivo de "símbolos" que inclui o símbolo NIL:

$$\left\{ \begin{array}{l}
GIC(Sym) \cong SYM \\
GIC(SetOfSyms) \cong 2^{SYM} \\
GIC(PairOfSetOfSyms) \cong SetOfSyms^2 \\
GIC(SeqOfSyms) \cong SYM^\star \\
GIC(Prod) \cong Sym \times SeqOfSyms \\
GIC(Grammar) \cong PairOfSetOfSyms \times Sym \times 2^{Prod} \\
GIC(Bool) \cong 2 \\
GIC(init) \stackrel{\text{def}}{=} \langle \emptyset, \emptyset, NIL, \emptyset \rangle \\
GIC(gramSym) \stackrel{\text{def}}{=} \lambda g.\pi_1(g) \\
GIC(ntermSym) \stackrel{\text{def}}{=} \lambda g.gramSym(g)(1) \\
GIC(termSym) \stackrel{\text{def}}{=} \lambda g.gramSym(g)(2) \\
GIC(axiom) \stackrel{\text{def}}{=} \lambda g.\pi_2(g) \\
GIC(prods) \stackrel{\text{def}}{=} \lambda g.\pi_3(g) \\
GIC(addNTSym) \stackrel{\text{def}}{=} \lambda(g,s).\langle gramSym(g) \dagger \begin{pmatrix} 1 \\ ntermSym(g) \cup \{s\} \end{pmatrix}, axiom(g), prods(g) \rangle \\
GIC(addTSym) \stackrel{\text{def}}{=} \lambda(g,s).\langle gramSym(g) \dagger \begin{pmatrix} 2 \\ termSym(g) \cup \{s\} \end{pmatrix}, axiom(g), prods(g) \rangle \\
GIC(addProd) \stackrel{\text{def}}{=} \lambda(g,p).\langle gramSym(g), axiom(g), prods(g) \cup \{p\} \rangle \\
GIC(isGIC) \stackrel{\text{def}}{=} \lambda g. \begin{array}[t]{ll} let & N = ntermSym(g) \\ & T = termSym(g) \\ & s = axiom(g) \\ & P = prods(g) \\ in & N \cap T = \emptyset \wedge s \in N \wedge \pi_1[P] \subseteq N \wedge \forall p \in P : elems(\pi_2(p)) \subseteq N \cup T \end{array}
\end{array} \right.$$

Após uma análise deste fragmento, construa a interface $IGIC$ por forma a GIC poder ser considerado de facto uma $IGIC$-álgebra, e descreva por palavras suas (breves!) o significado em GIC de cada operador de $IGIC$.

Questão 6 O texto que é dado em anexo — (12) na página 6 — é um fragmento, muito simplificado, dos requisitos informais colocados por uma empresa comercial a uma empresa informática que a primeira contratou para lhe produzir determinado ‘software’.

O fragmento de especificação SETS que se lhe segue, também em anexo — (13) na página 6 — é um esboço de modelo de dados para os requisitos propostos, em que todas as espécies não definidas devem ser encaradas como atómicas.

Após uma breve análise dos requisitos (12) e fragmento de modelo (13), construa a especificação das seguintes operações sobre um modelo cujo estado interno é $System$:

1. Operação
$$NULL\_INV : IdL \longrightarrow 2^{InvNr}$$
que deverá indicar os úmeros de todas as facturas anuladas numa dada loja.
2. Operação
$$SALES\_BY\_DATE : Date \longrightarrow (IdA \hookrightarrow \mathbb{N})$$
que deverá totalizar as vendas de um dado dia (em todas as lojas).

Questão 7 Considere o seguinte fragmento de gramática de contexto livre para expressões de aritmética inteira,

$$\begin{array}{rcl} G & = & \langle NT, T, \langle \mathit{Exp} \rangle, P \rangle \\ NT & = & \{ \langle \mathit{Exp} \rangle, \langle \mathit{Int} \rangle, \langle \mathit{BinOp} \rangle \} \\ T & = & \{ \texttt{+} , \texttt{-} , \texttt{*} , \texttt{/} , \texttt{(} , \texttt{)} , \ldots , \texttt{-2} , \texttt{-1} , \texttt{0} , \texttt{1} , \texttt{2} , \ldots \} \\ P & = & \left\{ \begin{array}{rcl} \langle \mathit{Exp} \rangle & ::= & \langle \mathit{Int} \rangle \mid \texttt{(} \langle \mathit{Exp} \rangle \texttt{)} \mid \langle \mathit{Exp} \rangle \langle \mathit{BinOp} \rangle \langle \mathit{Exp} \rangle \\ \langle \mathit{Int} \rangle & ::= & \ldots \texttt{-2} \mid \texttt{-1} \mid \texttt{0} \mid \texttt{1} \mid \texttt{2} \mid \ldots \\ \langle \mathit{BinOp} \rangle & ::= & \texttt{+} \mid \texttt{-} \mid \texttt{*} \mid \texttt{/} \end{array} \right. \end{array}$$

que é possível converter no seguinte sistema de equações em SETS:

$$\left\{ \begin{array}{rcl} Exp & \cong & Int + Exp + 4 \times Exp^2 \\ Int & \cong & \mathbb{Z}_0 \end{array} \right. \tag{2}$$

Considere o seguinte desafio: representar expressões aritméticas de $\langle \mathit{Exp} \rangle$ num tipo de dados em PASCAL que não recorra a ‘records’ variantes. Com base em (2), mostre que a seguinte proposta de um colega seu,

```
type Exp = record
    L: ^Node;
    I: Integer
   end;
Node = record
    P: ^BinOps;
    N: ^Node;
    end;
BinOps = record
      O: BinOp;
      E: Exp
      end;
BinOp = (TIMES,DIV,PLUS,DIFF);
```

está correcta, onde `Integer` representa o não-terminal $\langle \mathit{Int} \rangle$. **Sugestão**: “inverta” o PASCAL dado, simplifique-o, e recorra ao resultado provado na Questão 2.

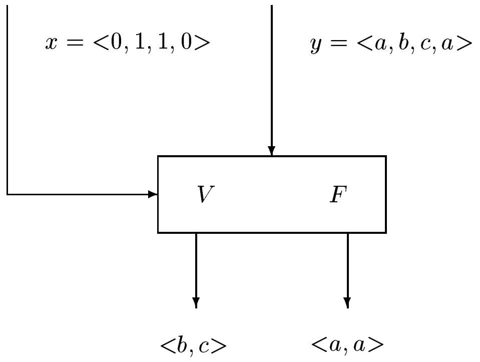
Questão 8 O facto

$$(A^\star)^2 \quad \trianglelefteq_f^\phi \quad 2^\star \times A^\star$$

verifica-se, em SETS, para mais do que uma possibilidade de f e ϕ.

Proponha-os por forma a que $f(\langle x, y\rangle)$ funcione da seguinte forma, ilustrada pelo diagrama abaixo: cada booleano em x actua como filtro, separando a sequência y nas duas sub-sequências que são dadas como resultado.

Exemplo, para $x = <0, 1, 1, 0>$ e $y = <a, b, c, a>$:



Será a função de abstracção que acaba de propôr injectiva? Justifique.

Questão 9 O texto que é dado em anexo — (12) na página 6 — é um fragmento, muito simplificado, dos requisitos informais colocados por uma empresa comercial a uma empresa informática que a primeira contratou para lhe produzir determinado ‘software’.

O fragmento de especificação SETS que se lhe segue, também em anexo — (13) na página 6 — é um esboço de modelo de dados para os requisitos propostos, em que todas as espécies não definidas devem ser encaradas como atómicas.

Após uma análise de requisitos (12) e fragmento de modelo (13), calcule a implementação relacional deste último usando o cálculo SETS, sintetizando o invariante de abstracção global.

Questão 10 No contexto do conhecido refinamento de conjuntos por sequências ($2^A \trianglelefteq_{elems} A^\star$), calcule a simulação do cálculo do segmento inicial de um número natural, isto é, resolva em ordem a x o seguinte diagrama de refinamento:

$$\begin{tikzcd} \mathbb{N}_0 \arrow[r, "\lambda n.\overline{n}"] & 2^{\mathbb{N}} \\ \mathbb{N}_0 \arrow[u, "id"] \arrow[r, "x"'] & \mathbb{N}^\star \arrow[u, "elems"'] \end{tikzcd}$$

Será o resultado do seu cálculo único? Justifique.

Questão 11 O conjunto dos números inteiros ($\mathbb{Z}_0$) é refinável em $\mathbb{N}_0 + \mathbb{N}_0$ (onde a etiqueta do co-produto “representa” o sinal, positivo ou negativo) sob a função de abstracção $f = id \mid sym$, onde se usa a construção funcional

$$\begin{array}{rcl} f \mid g & : & A + B \longrightarrow C \\ f \mid g(x) & \stackrel{\rm def}{=} & \left\{ \begin{array}{lcl} x = i_1(a) & \Rightarrow & f(a) \\ x = i_2(b) & \Rightarrow & g(b) \end{array} \right. \end{array}$$

e onde $sym(n) \stackrel{\rm def}{=} -n$.

Considere, neste contexto, o diagrama de refinamento de função que calcula o quadrado de um inteiro, $sq(z) \stackrel{\rm def}{=} z^2$, onde α é a variável:

$$
\begin{tikzcd}
\mathbb{Z}_0 \arrow[r, "sq"] & \mathbb{Z}_0 \\
\mathbb{N}_0 + \mathbb{N}_0 \arrow[u, "id \mid sym"] \arrow[r, "\alpha"'] & \mathbb{N}_0 + \mathbb{N}_0 \arrow[u, "id \mid sym"']
\end{tikzcd}
$$

1. Justifique os seguintes passos do cálculo:

$$
\begin{aligned}
(id \mid sym) \circ \alpha &= sq \circ (id \mid sym) && (3)\\
&= sq \mid (sq \circ sym) && (4)\\
&= sq \mid sq && (5)\\
&= id \circ (sq \mid sq) && (6)\\
&= (id \mid sym) \circ i_1 \circ (sq \mid sq) && (7)
\end{aligned}
$$

Nota: baseie-se, entre outros, nos factos:

$$
\begin{aligned}
h \circ (f \mid g) &= (h \circ f) \mid (h \circ g) && (8)\\
(f \mid g) \circ i_1 &= f && (9)\\
(f \mid g) \circ i_2 &= g && (10)
\end{aligned}
$$

2. Complete o processo de cálculo por forma a obter, finalmente

$$
\alpha(x) \stackrel{\rm def}{=} \left\{ \begin{array}{rcl} x = i_1(a) & \Rightarrow & i_1(a^2) \\ x = i_2(b) & \Rightarrow & i_1(b^2) \end{array} \right.
$$

Questão 12 Considere a seguinte função

$$
\begin{array}{rcl}
g & : & Y \longrightarrow M \\
g(y) & \stackrel{\rm def}{=} & \left\{ \begin{array}{rcl} p(y) & \Rightarrow & v \\ \neg(p(y)) & \Rightarrow & d(y)\, \theta\, g(e(y)) \end{array} \right.
\end{array}
$$

onde θ é associativa e v é um qualquer elemento de M.

1. Mostre que, para todo o $y \in Y$, a igualdade

$$
g(y) = f(y)\, \theta\, v \qquad (11)
$$

se verifica, onde

$$
\begin{array}{rcl}
f & : & Y \longrightarrow M \\
f(y) & \stackrel{\rm def}{=} & \left\{ \begin{array}{rcl} p(y) & \Rightarrow & u \\ \neg(p(y)) & \Rightarrow & d(y)\, \theta\, f(e(y)) \end{array} \right.
\end{array}
$$

e u é elemento neutro de θ.

2. Mostre que, desde que θ seja também comutativa, então g é concretizável algoritmicamente no seguinte ciclo-`while`:

```
{   M r = v;
    Y yy = y;
    while (¬ p(yy))
       {   r = r θ d(yy);
           yy = e(yy);
       };
}
```

registado na notação "pseudo-C" usada nesta disciplina.
Sugestão: recorde a definição da função $floop$ (9.28) apresentada na sua sebenta na página 347.

3. No caso de v ser o elemento absorvente de θ, que pode dizer do comportamento do ciclo-`while` acima? Justifique formalmente.

Anexo

Requisitos:

(...) A empresa XYZ LDA. *consta de 3 estabelecimentos ou lojas de comércio, prevendo-se que este número venha a aumentar no futuro.* *A informação associada a cada loja consta de (a) o 'stock' de artigos armazenados ou expostos para venda (cada artigo tem um código próprio); (b) as folhas de vendas.* *Existe uma folha de venda por cada dia útil. Cada folha de venda regista as facturas emitidas nas vendas daquele dia. Cada factura indica os artigos vendidos e em que quantidade. As facturas são numeradas e emitidas sequencialmente. Pode haver facturas anuladas. (...)*

(12)

Modelo (esboço):

$$\left\{ \begin{array}{rcl} System & \cong & IdL \hookrightarrow Info \\ Info & \cong & Stock \times Sales \\ Stock & \cong & IdA \hookrightarrow \mathbb{N} \\ Sales & \cong & InvNr \hookrightarrow (Date \times (IdA \hookrightarrow \mathbb{N})) \\ InvNr & \cong & \mathbb{N} \text{ /*\textit{invoice number} */} \end{array} \right. \tag{13}$$