# Universidade do Minho

| 2006/07 | 1.º Semestre ☐ 2.º Semestre ☒ Anual ☐ |
|---|---|
| DISCIPLINA Cálculo de Programas (8504N1)<br>CURSO LCC | DOCENTE J.N. Oliveira – 406006 |

| AULA | SUMÁRIO |
|---|---|
| Teórica<br>2007.02.26<br>2.ª-feira, 15h00-16h00<br>Sala CP2-104 | Apresentação da disciplina. Equipa docente.<br>Programa da disciplina e seu enquadramento no plano de estudos. Motivação. Teoria e método em programação. Composicionalidade. Breve introdução à Programação funcional. Sua motivação e antecendentes históricos.<br>Regime de avaliação. Bibliografia. Informação electrónica sobre a disciplina: `URL: http://www.di.uminho.pt/~jno/html/cp.html`.<br><br>O DOCENTE ______ |

| AULA | SUMÁRIO |
|---|---|
| Teórica<br>2007.03.01<br>5.ª-feira, 16h00-17h00<br>Sala CP1 A5 | Análise de requisitos e sua captação funcional. Exemplo: gestão de listas de chamadas num telemóvel. Concepção composicional e reutilização. Representação de funções por diagramas. Domínio e codomínio de uma função. Setas $A \xrightarrow{f} B$. Notação funcional com ou sem variáveis.<br>*Início do estudo dos combinadores de programas funcionais:* A composição $f \cdot g$ como combinador elementar de funções. Associatividade da composição: $f \cdot (g \cdot h) = (f \cdot g) \cdot h$ (2.8).<br>Função identidade $id$. O polimorfismo de $id$ e a propriedade $f \cdot id = id \cdot f = f$ e seu diagramas comutativo (2.10). Breves comentários sobre o *polimorfismo* de $id$.<br><br>O DOCENTE ______ |

| AULA | SUMÁRIO |
|---|---|
| Teórica<br>2007.03.05<br>2.ª-feira, 15h00-16h00<br>Sala CP2-104 | *Estudo dos combinadores de programas funcionais (cont.):* O combinador $\langle f, g\rangle$ e o produto $A \times B$ (analogia com "`struct`" em C) e suas projecções. Sua definição *pointwise*. Uso de diagramas para inferir propriedades. Exemplo: propriedade de cancelamento-$\times$ (2.20). Uso de diagramas para inferir operadores. Exemplo: o produto de funções $f \times g$ (2.22).<br><br>O DOCENTE ______ |

| AULA | SUMÁRIO |
|---|---|
| Teórica<br>2007.03.08<br>5.ª-feira, 16h00-17h00<br>Sala CP1 A5 | *Estudo dos combinadores de programas funcionais (cont.):* Propriedade de fusão-$\times$ (2.24) e de absorção-$\times$ (2.25). Apresentação da extensão `Cp.hs` ao `Prelude.hs` do Haskell.<br><br>O DOCENTE ____________________ |

| AULA | SUMÁRIO |
|---|---|
| Teórica<br>2007.03.12<br>2.ª-feira, 15h00-16h00<br>Sala CP2-104 | *Estudo dos combinadores de programas funcionais (cont.):* O combinador $[f\ ,g]$ e o coproduto $A+B$ (analogia com "`union`" em C) e suas injecções. Propriedades universais de $\langle f, g\rangle$ (2.55) e de $[f\ ,g]$ (2.57). Propriedades de reflexão-$\times$ (2.30) e reflexão-$+$ (2.39). Propriedades de fusão-$+$ (2.40), de cancelamento-$\times$ (2.20) e de cancelamento-$+$ (2.38).<br><br>O DOCENTE ____________________ |

| AULA | SUMÁRIO |
|---|---|
| Teórica<br>2007.03.15<br>5.ª-feira, 16h00-17h00<br>Sala CP1 A5 | *Estudo dos combinadores de programas funcionais (cont.):* Introdução à representação de predicados por guardas (2.60). Combinador condicional de McCarthy (2.59). Enunciado das leis de fusão do condicional de McCarthy (2.61,2.63) e demonstração da primeira destas.<br>Introdução à noção de isomorfismo entre tipos de dados. Motivação: a função $swap = \langle \pi_2, \pi_1 \rangle$, sua propriedade involutiva ($swap \cdot swap = id$) e o isomorfismo $A \times B \cong B \times A$ (2.31).<br>Funções bijectivas ou isomorfismos. Funções conversas.<br>Tipos elementares genéricos: $0$, $1$ e $2$ (resp. `Void`, `()` e `Bool` em HASKELL) e seus isomorfismos básicos: $A \times 1 \cong 1$ (2.81), $A + 0 \cong 0$ (2.79) e $A \times 0 \cong 0$ (2.80).<br><br>O DOCENTE ____________________ |

| AULA | SUMÁRIO |
|---|---|
| Teórica<br>2007.03.19<br>2.ª-feira, 15h00-16h00<br>Sala CP2-104 | *Estudo dos combinadores de programas funcionais (cont.):* A função $! : A \longrightarrow 1$. Funções constantes. O combinador $\underline{c}$. Propriedades. Polimorfismo da função constante: $\underline{c} = \underline{c} \cdot f$. O tipo de dados $1 + A$ ("apontador" para valores de tipo $A$). O combinador $f + g$. Propriedades functoriais do produto (2.28,2.29) e do coproduto (2.42,2.43).<br>Demonstração da animação destes combinadores na biblioteca `Cp.hs`.<br><br>O DOCENTE ____________________ |

| AULA | SUMÁRIO |
|---|---|
| Teórica<br>2007.03.22<br>5.ª-feira, 16h00-17h00<br>Sala CP1 A5 | Lei da troca (2.47). Diagrama da lei da troca. Síntese do isomorfismo $undistr$ (2.49) que testemunha $(A \times B) + (A \times C) \cong A \times (B + C)$.<br>*Início do estudo de combinadores envolvendo tipos indutivos:* Recurso ao tipo de dados $1+A$ para para a modelação de listas ligadas. A equação $L \cong 1+A \times L$ (3.4). Discussão sobre a sua "resolução em ordem a $L$" quando comparada com a de uma equação algébrica convencional. A necessidade de encontrar soluções "a menos de um isomorfismo" (3.6).<br><br>O DOCENTE ____________ |

| AULA | SUMÁRIO |
|---|---|
| Teórica<br>2007.03.26<br>2.ª-feira, 15h00-16h00<br>Sala CP2-104 | Não houve aula (envolvimento do docente na organização da conferência ETAPS'07).<br><br>O DOCENTE ____________ |

| AULA | SUMÁRIO |
|---|---|
| Teórica<br>2007.03.29<br>5.ª-feira, 16h00-17h00<br>Sala CP1 A5 | Não houve aula (envolvimento do docente na organização da conferência ETAPS'07).<br><br>O DOCENTE ____________ |

| AULA | SUMÁRIO |
|---|---|
| Teórica<br>2007.04.12<br>5.ª-feira, 16h00-17h00<br>Sala CP1 A5 | *Estudo de combinadores envolvendo tipos indutivos (cont.) :* Estudo das funções que observam o (isto é, têm à entrada valores do) tipo $A^\star$. Noção de *catamorfismo* de tipo $A^\star$ e sua analogia com o combinador `foldr` do Haskell. Diagrama de um catamorfismo.<br><br>O DOCENTE ____________ |

| AULA | SUMÁRIO |
|---|---|
| Teórica<br>2007.04.16<br>2.ª-feira, 15h00-16h00<br>Sala CP2-104 | *Estudo de combinadores envolvendo tipos indutivos (cont.) :* Propriedade universal do combinador *catamorfismo de listas*:<br>$$k = (\!\|g\|\!) \quad \Leftrightarrow \quad k \cdot in = g \cdot \mathsf{F}\, k \qquad (1)$$<br>onde $\mathsf{F}\, k = id + id \times k$ e $in, out$ foram deduzidas na aula anterior.<br>Dedução, a partir da lei anterior, das propriedades de reflexão e cancelamento-cata: *(v.s.f.f.)* |

| | |
|---|---|
| *(cont.)* | $$\begin{aligned} (\!| in |\!) &= id & (2) \\ (\!| g |\!) \cdot in &= g \cdot \mathsf{F}\,(\!| g |\!) & (3) \end{aligned}$$ O DOCENTE ____________ |

| AULA | SUMÁRIO |
|---|---|
| Teórica<br>2007.04.19<br>5.ª-feira, 16h00-17h00<br>Sala CP1 A5 | *Estudo de combinadores envolvendo tipos indutivos (cont.) :* A lei de recursividade múltipla (3.74) (também conhecida por lei de Fokkinga) e do seu corolário mais útil, a lei de "banana split" (3.76).<br>Aplicação: intercombinação de "ciclos" por fusão (horizontal). Exemplo: fusão dos cálculos da soma e do comprimento de uma lista no cálculo de uma média.<br><br>O DOCENTE ____________ |

| AULA | SUMÁRIO |
|---|---|
| Teórica<br>2007.04.23<br>2.ª-feira, 15h00-16h00<br>Sala CP2-104 | *Estudo de combinadores envolvendo tipos indutivos (cont.) :* Derivação da lei de *fusão-cata* (3.61): $$f \cdot (\!| g |\!) \quad = \quad (\!| \beta |\!) \quad \text{se} \quad f \cdot g = \beta \cdot \mathsf{F}\, f$$ Exemplo de aplicação: cálculo da soma dos quadrados dos elementos de uma lista<br><br>O DOCENTE ____________ |

| AULA | SUMÁRIO |
|---|---|
| Teórica<br>2007.04.27<br>5.ª-feira, 16h00-17h00<br>Sala CP1 A5 | *Estudo dos combinadores de programas funcionais (cont.) :* Noção de *anamorfismo* de listas (3.34) e sua analogia com o combinador `unfold`. Noção de *hilomorfismo* e princípio da *hilo-factorização* algorítmica. Hilomorfismos de listas (3.35). Exemplo: a função factorial.<br><br>O DOCENTE ____________ |

| AULA | SUMÁRIO |
|---|---|
| Teórica<br>2007.04.30<br>2.ª-feira, 15h00-16h00<br>Sala CP2-104 | Apresentação do módulo `List.hs`. A triologia *cata-ana-hilo* associada ao tipo listas em Haskell e exemplos de aplicação.<br>Generalização dos conceitos de cata/ana/hilo-morfismos a outros tipos de dados. Costumização de produtos e coprodutos em HASKELL, cf. eg. (2.93). A dualidade de isomorfismos $in/out$ associados à cláusula `data` do Haskell. Álgebras e coálgebras de tipos de dados. Interpretação das declarações `data` do Haskell. A álgebra $in$ e sua inversa $out$. Transformada "pointfree" aplicada a funções recursivas.<br><br>O DOCENTE ______________ |

| AULA | SUMÁRIO |
|---|---|
| Teórica<br>2007.05.03<br>5.ª-feira, 16h00-17h00<br>Sala CP1 A5 | Generalização dos conceitos de cata/ana/hilo-morfismos a outros tipos de dados. Primeiro exemplo: árvores binárias.<br>Apresentação do módulo `BTree.hs` Estudo da triologia *cata-ana-hilo* associada ao tipo `BTree`. Exemplo: o hilomorfismo `qSort` ('quick sort').<br><br>O DOCENTE ______________ |

| AULA | SUMÁRIO |
|---|---|
| Teórica<br>2007.05.07<br>2.ª-feira, 15h00-16h00<br>Sala CP2-104 | *Apresentação do módulo* `LTree.hs` *:* Estudo da triologia *cata-ana-hilo* associada ao tipo `LTree`. Exemplos: o hilomorfismo `mSort` ('merge sort') e sua comparação com `ISort` e `qSort` como exemplo de transferência de carga algorítmica entre os genes de um hilomorfismo. Os hilomorfismos `dfac` (*duplo factorial*) e `fib` (*série de Fibonacci*).<br><br>O DOCENTE ______________ |

| AULA | SUMÁRIO |
|---|---|
| Teórica<br>2007.05.10<br>5.ª-feira, 16h00-17h00<br>Sala CP1 A5 | Não houve aula (tolerância das JOIN'07).<br><br>O DOCENTE ______________ |

| AULA | SUMÁRIO |
|---|---|
| Teórica<br>2007.05.14<br>2.ª-feira, 15h00-16h00<br>Sala CP2-104 | Não houve aula (tolerância do Enterro da Gata).<br><br>O DOCENTE ______________ |

| AULA | SUMÁRIO |
|---|---|
| Teórica<br>2007.05.17<br>5.ª-feira, 16h00-17h00<br>Sala CP1 A5 | Não houve aula (tolerância do Enterro da Gata).<br><br>O DOCENTE ____________________ |

| AULA | SUMÁRIO |
|---|---|
| Teórica<br>2007.05.21<br>2.ª-feira, 15h00-16h00<br>Sala CP2-104 | *Parametrização e polimorfismo — tipos de dados como functores :* Introdução ao conceito de *functor de tipo* ('type functor'). Síntese de `fmap` para o tipo `LTree` como um catamorfismo. Repetição do exercício anterior para `BTree`. Generalização: noção de functor. Propriedades functoriais — preservação da identidade (3.44) e da composição (3.45).<br><br>O DOCENTE ____________________ |

<table>
<tr><th>AULA</th><th>SUMÁRIO</th></tr>
<tr><td>Teórica<br>2007.05.24<br>5.ª-feira, 16h00-17h00<br>Sala CP1 A5</td><td>Parametrização e polimorfismo (conclusão) : Noção de bi-functor. Propriedades (3.46,3.47). Bi-functores em HASKELL: a class <code>BiFunctor</code> e o operador <code>bmap</code>. Exemplos: bifunctores produto e coproduto. Functores polinomiais. Definição genérica de um tipo indutivo de dados. Noção de functor de base. Operadores <code>fmap</code> vs catamorfismos: Politipismo da definição $T\ a \cong B(a, T\ a)$ de um tipo indutivo genérico paramétrico. Noção de functor de tipo e sua formulação genérica como o catamorfismo $T\ f \stackrel{\mathrm{def}}{=} (\!| in \cdot B(f, id) |\!)$ (3.66). Classificação algorítmica. Catálogo de tipos polinomiais indutivos (3.68). Quadro sinóptico dos principais algoritmos analisados e estudados ao longo da disciplina:
<table>
<tr><th>Classe</th><th>B(A,X)</th><th>Serialização</th><th>Ordenação</th><th>Inversão</th><th>Factorial</th><th>Quadrado</th><th>Outros</th></tr>
<tr><td>RList</td><td>$1 + A \times X$</td><td>$cRList2h$</td><td>$iSort$</td><td>$invl$</td><td>$fac$</td><td>$sq$</td><td>$look$</td></tr>
<tr><td>BTree</td><td>$1 + A \times X^2$</td><td>in/pré/pós</td><td>$qSort$</td><td></td><td></td><td></td><td>$hanoi, traces$</td></tr>
<tr><td>LTree</td><td>$A + X^2$</td><td>$tips$</td><td>$mSort$</td><td>$invLTree$</td><td>$dfac$</td><td>$dsq$</td><td>$fib$</td></tr>
</table>
O DOCENTE ____________________</td></tr>
</table>

<table>
<tr><th>AULA</th><th>SUMÁRIO</th></tr>
<tr><td>Teórica<br>2007.05.28<br>2.ª-feira, 15h00-16h00<br>Sala CP2-104</td><td>

*Composição funcional* ***monádica*** *:* Multi-funções (funções que dão listas como resultado) e sua composição. Definição da composição $f \bullet g$ em ambos os casos (4.1,4.3). Generalização: functores que são mónadas. Composição monádica (4.4) em geral:

$$f \bullet g \stackrel{\mathrm{def}}{=} \mu \cdot \mathsf{F}\, f \cdot g$$

$$\begin{tikzcd} \mathsf{F}(\mathsf{F}\, C) \arrow[d, "\mu"'] & \mathsf{F}\, B \arrow[l, "\mathsf{F}\, f"'] \arrow[d, dash, dotted] & A \arrow[l, "g"'] \\ \mathsf{F}\, C & B \arrow[l, "f"'] & \end{tikzcd}$$

Os operadores $\mu$ e $u$ e seus axiomas (4.5,4.6).

O DOCENTE ____________________

</td></tr>
</table>

<table>
<tr><th>AULA</th><th>SUMÁRIO</th></tr>
<tr><td>Teórica<br>2007.05.31<br>5.ª-feira, 16h00-17h00<br>Sala CP1 A5</td><td>

Mónadas versus functores. Mónadas em HASKELL— a class `Monad` e os operadores `return` e `(>>=)` (4.16). O operador `(>>)` e a notação `do` (4.18). Exemplos: listas e `Maybe`. Notação em compreensão (4.21). Uso da notação-`do` para exprimir propriedades, eg.

$$do\ \{\ a \leftarrow x\ ;\ return(f\ a)\ \} \quad = \quad (\mathsf{F}\, f)\ x \tag{4}$$

válido para toda a mónada $\mathsf{F}$.

O DOCENTE ____________________

</td></tr>
</table>

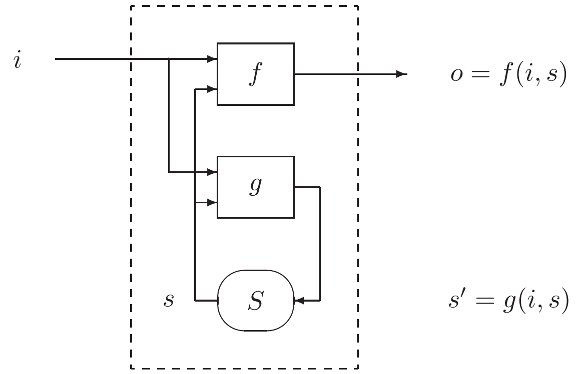
<table>
<tr><th>AULA</th><th>SUMÁRIO</th></tr>
<tr><td>Teórica<br>2007.06.04<br>2.ª-feira, 15h00-16h00<br>Sala CP2-104</td><td>

*A mónada de estado e suas transformações :* Noção de serviço em informática. Automação de serviços. Exemplos: uma caixa multibanco; uma base de dados; um *stack*. Noção de estado interno (base de dados) de um serviço.
A função de transição de um autómato determinístico vista como um “split” de duas funções, uma ($g$) que altera o estado interno ($s$) e outra ($f$) que devolve o resultado:

*(v.s.f.f.)*

</td></tr>
</table>

*(cont.)*




A mónada de (transição de) estado

$$\mathsf{ST}\ A \quad = \quad S \longrightarrow (A \times S) \tag{5}$$

e sua utilização para modelar a transição de estado de um autómato, por exemplo

$$pop() \quad = \quad \langle head, tail \rangle \quad \text{(para estados não vazios)} \tag{6}$$
$$push\ n \quad = \quad \langle \underline{ok}, (n :) \rangle \tag{7}$$

Componentes da mónada de estado:

$$u\ a \quad = \quad \langle \underline{a}, id \rangle \tag{8}$$
$$\mu\ \langle f, g \rangle \quad = \quad f' \cdot g \quad \text{onde } f'\ s = (f\ s)s \tag{9}$$

Estudo do ficheiro `SMonad.hs` do material pedagógico. Combinação de mónadas: transformadores de mónadas. Mónada de estado transformada por outra mónada M:

$$\mathsf{STM}\ A \quad = \quad S \longrightarrow \mathsf{M}(A \times S) \tag{10}$$

Exemplo: computações com estado e IO: modelo típico de um autómato determinístico interactivo (ficheiro `SIMonad.hs` do material pedagógico).
Projecto de software "por camadas": a camada puramente funcional, a camada reactiva e a camada interactiva. Exemplo de aplicação: serviço de gestão de listas de chamadas num telemóvel (ficheiro `mobile.hs` do material pedagógico).
Preenchimento do questionário de avaliação da disciplina.

O DOCENTE ______________________