

Métodos de Programação I

2.º Ano da LMCC (701055) + LESI (531316)

Ano Lectivo de 2000/2001

Exame (época normal. 1.ª chamada) – 11/1/2001

9:30

Salas 2206 a 2211

Grupo I

1. Mostre que se $f : A \rightarrow B$ é um isomorfismo então f^C também o é.
2. Use o resultado da alínea anterior para mostrar que $(A \times B)^C \cong (B \times A)^C$.
3. Demonstre a lei da *fusão da exponenciação* partindo da propriedade universal respectiva.

Grupo II

1. Considere a seguinte definição duma função que testa se uma lista contém elementos repetidos:

```
dups :: (Eq a) => [a] -> Bool
dups [] = False
dups (h:t) = (elem h t) || (dups t)
```

Explique por palavras suas porque é que esta função não pode ser expressa como um catamorfismo sobre listas. Defina-a como um hilomorfismo.

2. Na sequência da alínea anterior, defina em Haskell o tipo de dados intermédio usado na definição do hilomorfismo. Mostre qual o termo desse tipo que é usado no cálculo de `dups [1,2,3,1,2,3]`
3. Considere a seguinte declaração de tipo em *Haskell*

```
data Nat = Zero | Succ Nat
```

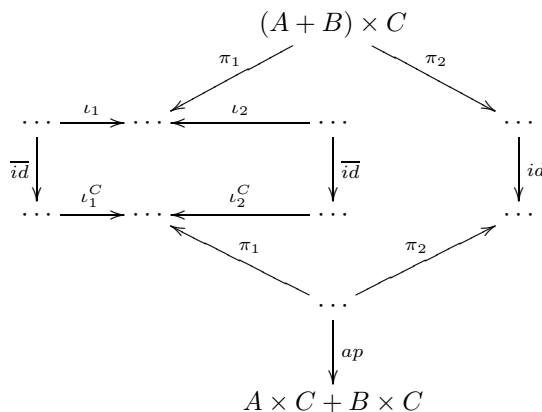
- (a) Defina as operações *cata/ana/hilo* para esse tipo de dados. Acompanhe essas definições com os diagramas respectivos.
- (b) O tipo apresentado é isomorfo ao tipo primitivo do *Haskell* `[]`. Apresente a definição em *Haskell* das funções que testemunham esse isomorfismo.
- (c) Defina a função `comp :: [a] -> Nat` (determina o comprimento de uma lista) como um anamorfismo do tipo `Nat`.

Grupo III

1. A função $distl : (A + B) \times C \rightarrow A \times C + B \times C$ (distributividade à esquerda) dispõe de uma definição *point-free* particularmente complicada:

$$distl = ap \cdot ([\iota_1^C \cdot \overline{id}, \iota_2^C \cdot \overline{id}] \times id)$$

- (a) Preencha o diagrama seguinte que justifica o tipo atribuído à função.



- (b) Conjecture a inversa $undistl$ e apresente diagramas que justifiquem o seu tipo.