

# Algoritmos y Estructuras de Datos I - 1º cuatrimestre 2010

## Práctico 4: Tipos abstractos de datos

1. Considere la función de abstracción  $\llbracket \cdot \rrbracket : (Nat, Nat) \rightarrow Int$  especificada como:

$$\llbracket (a, b) \rrbracket = a - b$$

que permite representar un número entero con un par de números naturales. Por ejemplo el número  $-17$  puede estar representado por  $(0, 17)$  (también por  $(1, 18)$ ,  $(2, 19)$ , etc.). Especifique formalmente y derive cada una de las siguientes funciones (los items marcados con  $*$  son algo más complicados):

- a)  $\oplus : (Nat, Nat) \rightarrow (Nat, Nat) \rightarrow (Nat, Nat)$ , que suma dos enteros representados por  $\llbracket \cdot \rrbracket$ .
- b)  $\ominus : (Nat, Nat) \rightarrow (Nat, Nat) \rightarrow (Nat, Nat)$ , que resta dos enteros.
- c)  $\otimes : (Nat, Nat) \rightarrow (Nat, Nat) \rightarrow (Nat, Nat)$ , que multiplica dos enteros.
- d)  $(*)\ominus : (Nat, Nat) \rightarrow (Nat, Nat) \rightarrow Bool$ , que compara con la relación de igualdad dos enteros.
- e)  $(*)\otimes : (Nat, Nat) \rightarrow (Nat, Nat) \rightarrow Bool$ , que compara con la relación  $<$  dos enteros.

2. Considere la función de abstracción  $\llbracket \cdot \rrbracket : [Nat] \rightarrow Nat$  especificada como:

$$\llbracket xs \rrbracket = \left\langle \sum i : 0 \leq i < \#xs : xs.i \times B^i \right\rangle$$

que permite representar (con algunas restricciones) un número natural dada una base  $B$  y una lista de números naturales  $xs$ . Por ejemplo el número 17 puede representarse en base 2 con la lista  $[1, 0, 0, 0, 1]$  (y aunque *no* es lo esperado también puede representarse en la misma base con la lista  $[8, 1]$ ).

- a) Demuestre que la siguiente es una definición recursiva para  $\llbracket \cdot \rrbracket$ :

$$\begin{aligned} \llbracket \cdot \rrbracket &\doteq 0 \\ \llbracket x \triangleright xs \rrbracket &\doteq x + B \times \llbracket xs \rrbracket \end{aligned}$$

- b) Derive una función  $\oplus : [Nat] \rightarrow [Nat] \rightarrow [Nat]$  que represente la suma de enteros representados con  $\llbracket \cdot \rrbracket$ , es decir, que satisfaga la especificación:  $\llbracket xs \oplus ys \rrbracket = \llbracket xs \rrbracket + \llbracket ys \rrbracket$ .
- c) En el ítem anterior Ud. no debería utilizar  $\llbracket \cdot \rrbracket$  como **función recursiva** en la definición que obtuvo. Sin embargo, en el proceso de derivación es válido utilizar de manera indiferente tanto la especificación de  $\llbracket \cdot \rrbracket$  como su definición recursiva, ¿por qué?
- d) Defina una función  $\otimes : [Nat] \rightarrow [Nat] \rightarrow [Nat]$  que represente la multiplicación de enteros, satisfaciendo la especificación:  $\llbracket xs \otimes ys \rrbracket = \llbracket xs \rrbracket \times \llbracket ys \rrbracket$ .

3. Considere la función de abstracción  $\llbracket \cdot \rrbracket : Nat \rightarrow Bool$  especificada como:

$$\llbracket n \rrbracket \equiv n \neq 0$$

Observe que  $\llbracket \cdot \rrbracket$  permite representar un valor *booleano* con un natural  $n$ . Por ejemplo para *true* podría haber varios candidatos, elija uno. Especifique formalmente y derive las siguientes funciones:

- a)  $\mathbb{F} : Nat$ , que representa a la constante *false*.
- b)  $\mathbb{T} : Nat$ , que representa a la constante *true*.
- c)  $\mathbb{\wedge} : Nat \rightarrow Nat \rightarrow Nat$ , que representa al operador  $\wedge$ .
- d)  $\mathbb{\vee} : Nat \rightarrow Nat \rightarrow Nat$ , que representa al operador  $\vee$ .
- e)  $\mathbb{\neg} : Nat \rightarrow Nat$ , que representa al operador  $\neg$ .

f)  $\oplus : Nat \rightarrow Nat \rightarrow Nat$ , que representa al operador  $\equiv$ .

4. Considere la función de abstracción  $\llbracket \cdot \rrbracket : [Bool] \rightarrow \{Nat\}$  (conjunto de naturales) especificada como:

$$\llbracket xs \rrbracket = \langle \bigcup i : 0 \leq i < \#xs \wedge xs.i : \{i\} \rangle$$

que permite representar un conjunto de números naturales con una lista de valores *booleanos*. Por ejemplo, el conjunto  $\{0, 1, 3, 5\}$  está representado por la lista  $[true, true, false, true, false, true]$ . **Lea atentamente las ayudas dadas al final del ejercicio.**

Especifique y derive las siguientes funciones:

- $\odot : [Bool] \rightarrow [Bool] \rightarrow [Bool]$ , que representa la unión de conjuntos habitual.
- $\oslash : [Bool] \rightarrow [Bool] \rightarrow [Bool]$ , que representa la intersección de conjuntos.
- $\ominus : [Bool] \rightarrow [Bool] \rightarrow [Bool]$ , que representa la diferencia de conjuntos.
- $\odot : Nat \rightarrow [Bool] \rightarrow Bool$ , que calcula cuando un número natural pertenece a un conjunto representado a través de  $\llbracket \cdot \rrbracket$ .

**Ayuda:**

- Generalice la especificación de  $\llbracket \cdot \rrbracket$ , obteniendo una función de abstracción  $\{\{\cdot\}\}_n$  que además de una lista tome un parámetro extra  $n$ , de modo que satisfaga  $\llbracket \cdot \rrbracket = \{\{\cdot\}\}_0$ .
- Obtenga una definición recursiva de  $\{\{\cdot\}\}_n$  que le permitirá realizar las cuentas de manera más simple y compacta (recuerde el ejercicio 2c).
- Las siguientes propiedades de conjuntos que pueden ser útiles:

$$\begin{array}{lll} (A \cup B) \cap C = (A \cap C) \cup (B \cap C) & A \cup \emptyset = A & A - \emptyset = A \\ (A \cup B) - C = (A - C) \cup (B - C) & A \cap \emptyset = \emptyset & \emptyset - A = \emptyset \\ A - (B \cup C) = (A - B) \cap (A - C) & & \end{array}$$