

TIPOS Y CLASES EN HASKELL

TIPOS

- Un tipo es una colección de valores relacionados
 - $v :: \tau$ v es un valor de tipo τ
 - $e :: \tau$ la expresión e produce un valor de tipo τ
- Cada expresión tiene un tipo que se calcula antes de ser evaluada mediante inferencia de tipos estática
 - $> :type\ e$ devuelve el tipo de la expresión e

TIPOS BÁSICOS

- **Bool** : True , False
- **Char** : 'a', '3', '\n', 'G',...
- **String** : "a3=B"
- **Int** : enteros con una cantidad de memoria fijada para almacenarlos
- **Integer** : enteros sin limitación de tamaño y precisión
- **Float** : números de coma flotante de precisión simple
- **Double** : coma flotante, admitiendo el doble de memoria de almacenamiento.

TIPOS COMPUESTOS

- *Tipo lista* : listas de elementos del mismo tipo $[\tau]$
- *Tipo tupla* : tuplas cuyas componentes pueden ser de distinto tipo,
() tupla de aridad 0, (τ_1, τ_2) par (tupla de aridad 2)
- *Tipo función* : $f :: \text{Int} \rightarrow \text{Int}$
- *Funciones currificadas* : funciones que se aplican a sus argumentos de uno en uno, permitiendo aplicación parcial
 - \rightarrow asocia por la derecha
 - La aplicación asocia por la izquierda
- *Tipos polimórficos* : variables de tipo (empiezan por minúscula)
- *Tipos sobrecargados* : Incluyen una restricción de clase
 $(+) :: \text{Num } a \Rightarrow a \rightarrow a \rightarrow a$ (si la variable a se instancia con un tipo de la clase Num ...)

CLASES

- Una clase es una colección de tipos que soportan operaciones sobrecargadas llamadas métodos.
- Clases básicas:
 - Eq
 - Ord
 - Show
 - Read
 - Num
 - Integral
 - Fractional

Clases Eq y Ord

- **Eq** : Tipos cuyos elementos pueden compararse c.r.a igualdad o desigualdad. Contiene a los tipos básicos
 - Métodos (`==`), (`/=`)
- **Ord** : Contiene a los tipos que son instancia de Eq, pero que además sus elementos puedan ordenarse
 - Métodos (`<`), (`>`), (`<=`), (`>=`), `min`, `max`

Clases Show y Read

- **Show** : Tipos cuyos elementos pueden convertirse en String o Char. Contiene a los tipos básicos; listas y tuplas si sus elementos son instancia de Show
 - Método `show :: a -> String`
- **Read** : Contiene a los tipos que son *legibles* Dual a Show (mismas instancias)
 - Método `read :: String -> a`

Clases Num, Integral y Fractional

- **Num** : Tipos cuyos elementos son numéricos.
 - Métodos (+), (-), (*), negate, abs, signum
- **Integral** : Tipos instancias de Num cuyos valores son enteros (Int, Integer)
 - Métodos div, mod
- **Fractional** : Tipos instancias de Num que no están en Integral (Float, Double)
 - Métodos (/), recip

Ejercicio: determinar el tipo de cada uno de los métodos presentados