

Paradigmas de Lenguajes de Programación

Tomás Agustín Hernández



Recordando Haskell

Para ejecutar un archivo hay que instalar GHCi. Una vez instalado, nos paramos en la terminal en el directorio donde está el archivo que queremos ejecutar.

- Cargar archivo: `:l nombreArchivo`
- Ver tipo: `:type tipo`
- Ejecutar funcion: `funcion parametro1 parametro2...`
- Recargar archivo: `:r`
- Si necesitamos hacer cálculos para mandar un parámetro, usar paréntesis: Ej.: `otherwise = n * factorial(n-1)`
-

Maybe

El Maybe se utiliza en Haskell para recibir/devolver respuestas condicionales que pueden ser de un tipo u otro.

Se define como *data Maybe a = Nothing | Just a*

Ej.: *devolverFalsoSiVerdadero : Bool → Prelude.Maybe Bool*

El Maybe deja la puerta abierta a un valor posible "Nothing". Entonces tenemos dos casos: Si me envían un True devuelvo False (tipo bool), caso contrario, devuelvo Nothing.

Either

El Either se utiliza en Haskell para poder recibir/devolver un parámetro que podría ser de un tipo u otro.

Se define como *data Either a b = Left a | Right b*

Para poder saber qué operación hacer según el tipo literalmente en código usamos (Left valor) o (Right valor).

Ej.: *devolverRepresentacionIntBool :: Either Int Bool → Int*

Si es un entero, devuelvo ese mismo entero porque no hago nada. Eso lo hacemos con *Left(a) = a*, ahora, si el tipo es booleano tengo que decir explícitamente la respuesta según su valor. Es decir, *Right(False) = 0* sino, *Right(True) = 1*.