

CI4251 - Programación Funcional Avanzada
Tarea 2

Ernesto Hernández-Novich
86-17791
<emhn@usb.ve>

Mayo 20, 2010

1. Transformador IdentityT (2 puntos)

Siguiendo la técnica descrita en clase, proponga una definición para el transformador `IdentityT`, que incluya todas las instancias razonables para operar como tal.

Sea un monad cualquiera M y su transformador asociado MT . ¿Será cierto o falso $IdentityTM == MTIdentity$?

2. Uso de Parsec (3 puntos)

Un archivo “Literate Haskell” (`.lhs`) incluye código Haskell combinado con texto arbitrario. A efectos de este ejercicio supondremos que se trata de texto simple pero se desea convertirlo a HTML para su publicación en una página Web. Más aún, se adoptan las siguientes convenciones:

- Si una línea comienza con `*` se trata de un encabezado principal.
- Si una línea comienza con `#` se trata de un encabezado secundario.
- Un párrafo termina cuando haya una línea en blanco.
- Si una línea comienza *exactamente* con `>` seguido de espacio en blanco, se asume que el resto corresponde a texto del programa. Los dos caracteres al principio **no** deben conservarse.
- Los espacios en blanco son irrelevantes salvo el caso anterior. En otras palabras, los espacios en blanco entre palabras, antes del `*` al comienzo de una línea, etc. han de convertirse en un espacio en blanco sencillo.

Escriba un programa basado en un reconocedor `Parsec` tal que pueda ser utilizado para convertir los archivos `.lhs` con el formato antes descrito hacia HTML válido, tomando en cuenta que:

- Los encabezados principales y secundarios deben envolverse entre las marcas HTML `<h1>` y `<h2>`.
- Los párrafos sueltos deben envolverse entre las marcas `<p>`.
- Los segmentos continuos (múltiples líneas seguidas) con código Haskell deben envolverse entre las marcas HTML `<code>`.
- Los símbolos `<`, `>` y `&` tienen significado especial en HTML, por lo que deben ser convertidos a la entidad correspondiente.

- El resto del texto debe ser transportado “tal cual”.

Su programa debe recibir uno o más nombres de archivo `.lhs` desde la línea de comandos y producir sendos archivos `.html` con los resultados de la transformación.