

UTN - FRBA	Paradigmas de Programación	Sábados turno mañana
Parcial	Funcional	30/05/2020



Siempre dicen que las crisis son oportunidades. Así que vamos a tomar esta oportunidad donde nadie quiere acciones, para crear una plataforma de compra y venta de acciones.

Tenemos que modelar las acciones de forma que podamos saber el símbolo que la identifica ("MCD" para McDonald's, por ejemplo) y el historial de precios del último año, teniendo en cuenta que el precio más actual es el primero, y el último el más antiguo.

Cada segundo pueden llegar actualizaciones en los precios de un mercado, es necesario que todas las acciones del mercado estén juntas para que se puedan ir actualizando. Así se representa.

```
data Accion = Accion {
    simbolo :: String,
    precios :: [Number]
} deriving Show
```

También tendremos usuarios, cada uno tendrá una cartera donde se encontrará el efectivo que tiene y una lista de títulos de acciones, que contienen el símbolo, la cantidad y precio al cual compró las acciones.

Implementar los siguientes requerimientos, maximizando la utilización de composición, aplicación parcial y orden superior. Tener en cuenta el orden de los parámetros de cada función de forma tal de facilitar la utilización de aplicación parcial.

Se prohíbe el uso de recursividad, salvo que se aclare lo contrario.

- 1) Generar los tipos y estructuras necesarios para modelar los usuarios y sus títulos.

Nota: No usar nombres de componentes repetidos en las distintas estructuras.

- 2) Implementar las funciones

- a) **mapCondicional:** Similar al map, recibe una función de transformación, pero además una condición, y la transformación de cada elemento se realiza únicamente si ese elemento cumple la condición dada. También recibe la lista a mapear, claro. Se espera que la lista resultante tenga la misma cantidad de elementos que la original. Escribir en un comentario: ¿de qué tipo tiene que ser la transformación para que esta función tenga sentido? ¿Por qué?

UTN - FRBA	Paradigmas de Programación	Sábados turno mañana
Parcial	Funcional	30/05/2020

- b) **encontrar**: Determina el primer elemento en una lista que cumple una condición.
- c) **cuantasTieneDe**: Dado un símbolo y un usuario, nos indica la cantidad de acciones que tiene ese usuario en sus títulos.

- 3)
- a) Crear una función **nuevoPrecioAccion** que reciba una acción y un nuevo precio y devuelva la acción con el precio actualizado.
 - b) Crear una función **nuevoPrecio** que reciba una lista de acciones, un símbolo de acción a actualizar y un nuevo precio y devuelva la lista de acciones con la acción a actualizar modificada.
 - c) Crear una función **precioActual** que, dada una acción, nos diga su precio actual.
- 4) Se sabe que a los usuarios les gusta saber su situación actual con un sólo número. Queremos una función **estadoActual** que reciba un usuario y una lista de acciones y devuelva la diferencia entre los precios de compra y los precios actuales para saber si está perdiendo o ganando.
- 5) Algunas acciones regularmente hacen pagos a sus accionistas para agradecer que todavía tengan sus acciones. Hacer una función **pagarDividendos** que reciba una lista de usuarios, un símbolo y la cantidad de dividendos que se da por acción y devuelva la lista de usuarios con el efectivo actualizado.
- 6) Como pequeño regalo, y para que no dejen de usar la plataforma, le queremos dar \$1.000 a aquellos usuarios a los que no les está yendo bien en un momento dado. Un usuario al que no le va bien es uno en el que su situación actual (punto 4) da una pérdida mayor a \$50.000. Hacer una función **rescateFinanciero** que reciba una lista de acciones y una lista de usuarios, y devuelva a los usuarios que cumplan el requerimiento con \$1.000 más.
- 7) Crear una función **venta** que reciba un usuario, una acción (no un símbolo) y una cantidad. Debe devolver al usuario con la transacción hecha. La transacción bajaría la cantidad de acciones y subiría el efectivo en ($\text{precioActual} * \text{cantidadAccionesVendidas}$).
- 8) Queremos mostrar la acción que más haya convenido comprar, para eso necesitamos de algunas funciones.
- a) Una función **porcentajeDeGanancia** que reciba una acción y devuelva el porcentaje que subió desde su primer medición hasta su precio actual (sería: $\text{precioActual} * 100 / \text{primerMedición}$)
 - b) Una función **mayorGanancia** que, dadas dos acciones, devuelva la acción con el porcentaje de suba más grande.
 - c) **Nota**: En este punto, se puede usar recursividad.
Una función **laMejorAccion** que dada una lista de acciones devuelva la acción con el mayor porcentaje de suba.
- 9) Explicar y justificar la inferencia del tipo de la siguiente función:
funcionQueNoDebeSerNombrada $x \ y = (>= \ x) \ . \ \text{foldr} \ (\text{flip} \ y \ 10) \ x$
Nota: No se pide sólo indicar el tipo. Se pide explicar de dónde surge cada uno de los tipos inferidos (parámetros y resultado).