

Máquinas de Helado Refurbished

TP Funcional - única entrega

Miércoles a la noche

Han pasado 8 años¹ desde el primer desarrollo de la máquina de helado que convirtió a “Helarte funcional” en un multimillonario holding con franquicias en todo el mundo, ofreciendo sus gustos como “Composición de sabores”, “Lambda al agua”, “Frutilla curryficada”, entre otros. Pero el negocio ha cambiado, así que nos pidieron que desarrollemos un nuevo sistema construido en Haskell que tenga las siguientes características.



Índice general

[1 Formato de entrega](#)

[2 Dominio](#)

[2.1 Punto 1 \(común\): Helados bien y mal preparados](#)

[Casos de prueba que debe definir](#)

[2.2 Punto 2](#)

[2.2.1 a. heladera \(común\)](#)

[Casos de prueba que debe definir](#)

[2.2.2 b. agregadora: común](#)

[Casos de prueba que debe definir](#)

[2.2.3 c. mixturadora: integrante 1](#)

[Casos de prueba que debe definir](#)

[2.2.4 d. batidora: común](#)

[Casos de prueba que debe definir](#)

[2.2.5 e. choripasteadora: integrante 2](#)

[Casos de prueba que debe definir](#)

[2.2.6 f. Abstraer el tipo de dato de una máquina : integrante 1](#)

[2.2.7 g. Helado neutro: integrante 1](#)

[2.3 Punto 3: Chocolate infinito - integrante 2](#)

[2.4 Punto 4](#)

[2.4.1 a. Cinta transportadora: integrante 2](#)

[2.4.2 b. Máquinas que preparan bien un helado : integrante 1](#)

[Casos de prueba que debe definir](#)

[2.4.3 c. BONUS Ingrediente favorito : común](#)

[Casos de prueba a definir](#)

¹ Ver [este examen](#)

1 Formato de entrega

Fecha de finalización: El assignment finaliza el miércoles 29/05 a las 18:00 hs. Pasado ese lapso el docente no verá los commits.

Cantidad de integrantes: deben formar grupos de a 2, en caso de quedar solos le asignaremos un compañero, salvo que efectivamente queden solos.

Nota: dentro del enunciado hay puntos específicamente sugeridos para cada uno de los integrantes (salvo los puntos que tengan “común” en su definición). En caso de haber un solo integrante (o que el otro integrante se de a la fuga), solo es obligatorio resolver una de las dos partes.

Defensa: la defensa del TP será presencial, el mismo miércoles 29/05 a partir de las 18:30. En caso de enfermedad o causa de fuerza mayor, debe enviar un mail a los tutores avisando el motivo de la ausencia y sugiriendo una fecha para defender el TP.

Qué debe incluir en el repositorio: el código fuente en Haskell que resuelva correctamente los puntos que se describen a continuación, la implementación de los casos de prueba exitosos, y en el README los integrantes del equipo y el *badge* contra Travis con el estado del último *build*. Todos los puntos anteriores más la aprobación de la defensa del TP permitirá que puedan presentarse a rendir el primer examen del Paradigma Funcional, de acuerdo a las reglas de juego que se comunicaron al principio del cuatrimestre.

2 Dominio

Todo helado tiene

- el nombre del gusto oficial, conformado por una cadena de caracteres que por convención están todas en minúscula
- la temperatura a la que está conservada
- y los ingredientes que tienen, modelados también con *strings*

```
type Ingrediente = String

data Helado = Helado {
  gusto    :: String,
  temperatura :: Int,
  ingredientes :: [Ingrediente]
} deriving (Show)
```

2.1 Punto 1 (común): Helados bien y mal preparados

Determinar si un helado está bien o mal preparado, lo que ocurre según estos parámetros

- la temperatura deberá estar al menos a -10 grados para cualquier helado que comience con chocolate (por ejemplo, para "chocolate intenso" o "chocolate nevado" sí, pero no para "vainilla al chocolate"). **Tip:** podés averiguar qué hace `Data.List.isPrefixOf`¹ y cómo la podés utilizar.
- si el helado tiene agua en sus ingredientes, debe estar al menos a -5 grados
- para el resto debe estar al menos a -x grados, donde x es la cantidad de ingredientes

Importante: tenés que usar composición en las funciones auxiliares

Casos de prueba que debe definir

Recomendación: genere valores constantes en el archivo de test ya que serán útiles en pruebas posteriores.

Condición	Qué se espera
Un helado de "chocolate intenso" a -22°C	Que esté bien preparado
Un helado de "chocolate intenso" a -10°C (valor límite)	Que esté bien preparado
Un helado de "chocolate intenso" a -2°C	Que no esté bien preparado
Un helado de "frutilla al agua" a -6°C con ingredientes "frutilla" y "agua"	Que esté bien preparado
Un helado de "frutilla al agua" a -5°C con ingredientes "frutilla" y "agua" (valor límite)	Que esté bien preparado
Un helado de "frutilla al agua" a -3°C con ingredientes "frutilla" y "agua"	Que no esté bien preparado
Un helado de "frutilla" a -3°C con ingredientes "frutilla" y "leche"	Que esté bien preparado
Un helado de "frutilla" a -2°C con ingredientes "frutilla" y "leche" (valor límite)	Que esté bien preparado
Un helado de "frutilla" a -1°C con ingredientes "frutilla" y "leche"	Que no esté bien preparado

1. Para importar `isPrefixOf` pueden hacer `import Data.List (isPrefixOf)`

2.2 Punto 2

Para poder producir el helado se necesita materia prima y máquinas que lo procesen. Modelar las siguientes máquinas y **dar el tipo de cada una de ellas:**

2.2.1 a. heladera (común)

Enfría una cierta cantidad de grados a un helado.

Casos de prueba que debe definir

Condición	Qué se espera
Utilizar la heladera para enfriar 5 grados la frutilla al agua que está a -6°C	Que la temperatura quede a -11°C
Utilizar la heladera para enfriar 8 grados el chocolate intenso que está a -2°C	Que pase a estar bien preparado

2.2.2 b. agregadora: común

Permite agregar un ingrediente a un helado al final de la lista.

Casos de prueba que debe definir

Condición	Qué se espera
Agregar "crema" al helado de chocolate bien preparado de -22°C .	Que tenga 4 ingredientes
Agregar "crema" al helado de chocolate bien preparado de -22°C .	Que el último ingrediente sea "crema"

2.2.3 c. mixturadora: integrante 1

Toma dos helados de frutas diferentes y obtiene uno nuevo, de gusto combinado concatenando " y " entre los gustos, con la menor temperatura y todos los ingredientes (sin repetidos).

Resolver la función que elimina repetidos en forma recursiva y llamarla *sinRepetidos*. Ésta es la única función que puede definir recursivamente.

Casos de prueba que debe definir

Condición	Qué se espera
Al mezclar un helado de frutilla al agua bien preparado a -6°C (con frutilla y agua, el	el gusto debe ser "frutilla al agua y durazno" (en ese orden ya que en ese orden

mismo del punto 1) y uno de durazno, a -2°C, con ingredientes durazno, agua y azúcar	se mezclan)
Al mezclar un helado de frutilla al agua bien preparado a -6°C (con frutilla y agua) y uno de durazno, a -2°C, con ingredientes durazno, agua y azúcar	la temperatura debe estar a -6°C
Al mezclar un helado de frutilla al agua bien preparado a -6°C (con frutilla y agua) y uno de durazno, a -2°C, con ingredientes durazno, agua y azúcar	los ingredientes deben ser ["agua", "azucar", "durazno", "frutilla"]. Consejo: usar la función sort <code>import Data.List (isPrefixOf, sort)</code>

2.2.4 d. batidora: común

Toma un cajón de una fruta, un dispenser de agua a una cierta temperatura, y un helado, transformándolo en un helado del sabor de la fruta del cajón, la temperatura del agua y los ingredientes agua y la fruta. Modelar el dispenser y el cajón como ud. lo crea conveniente

Casos de prueba que debe definir

Condición	Qué se espera
Al batir un cajón de "banana" con un dispenser a -4°C a un helado neutro	el gusto debe ser "banana"
Al batir un cajón de banana con un dispenser a -4°C a un helado neutro	la temperatura debe estar a -4°C
Al batir un cajón de banana con un dispenser a -4°C a un helado neutro	los ingredientes deben ser ["agua", "banana"]. Consejo: usar la función sort <code>import Data.List (isPrefixOf, sort)</code>

El helado neutro es parte del desarrollo del punto 2.2.7g).

2.2.5 e. choripasteadora: integrante 2

Es una máquina especial que le agrega "de la casa" al gusto del helado y también le incorpora los ingredientes "agua" y "esencia artificial", siempre y cuando no lo tengan.

Casos de prueba que debe definir

Condición	Qué se espera
Al choripastear el helado de chocolate intenso bien preparado (a -22°C)	el gusto debe decir "chocolate intenso de la casa"

Al choripastear el helado de chocolate intenso bien preparado (a -22°C)	la lista de ingredientes debe ser ["agua", "chocolate", "esencia artificial", "leche", "nesquik"]. Consejo: usar la función <code>sort</code> <code>import Data.List (isPrefixOf, sort)</code>
---	--

2.2.6 f. Abstraer el tipo de dato de una máquina : integrante 1

```
type Maquina = ...
```

2.2.7 g. Helado neutro: integrante 1

Definir un helado neutro, que no tiene gusto ni ingredientes, y está a 0 grados (es la base para las preparaciones)

Mediante el uso de composición, mostrar cómo podría tener un helado neutro, batirlo con frutilla y un dispenser de -5 grados, pasarlo por una heladera que le baje 5 grados, y agregarle azúcar. Explique dónde aparece el concepto de aplicación parcial.

2.3 Punto 3: Chocolate infinito - integrante 2

Definir un helado de Chocolate infinito, a -4 grados, que tiene como ingredientes azúcar, leche e infinitas veces chocolate. **Tip:** usar la función `repeat`.

¿Es posible invocar a la función heladera para que enfríe 2 grados con el chocolate infinito y verificar a qué temperatura queda? En caso afirmativo, desarrollar un test, justificar y relacionar con un concepto.

¿Es posible combinar el chocolate infinito con un helado de frutilla? Justificar.

2.4 Punto 4

2.4.1 a. Cinta transportadora: integrante 2

Generar una cinta transportadora que se conforma con un conjunto de máquinas y arma la preparación que resulta de tomar el helado neutro y aplicarle todas las máquinas.

Construya varios tests que muestre cómo resolver el mismo problema que en el punto 2.g con la cinta transportadora, es decir que al helado neutro le aplique

- la batidora de frutilla con el dispenser a -5°C
- la heladera que enfríe 5 grados
- y le agregue el ingrediente "azucar"

Debe verificarse (en diferentes tests)

- que la temperatura final quede a -10°C
- que los ingredientes finales sean ["agua", "frutilla", "azucar"]

Indique dónde aparece el concepto de aplicación parcial.

No se puede utilizar recursividad en ningún paso de este punto.

2.4.2 b. Máquinas que preparan bien un helado : integrante 1

Dada esta lista de máquinas:

1. Una que agrega "frutilla"
2. Otra heladera que enfría 5 grados
3. Otra heladera que calienta 10 grados
4. Una batidora de ananá y un dispenser de -25°C
5. Y una mixturadora de helado neutro

Queremos saber qué máquinas dejarían bien preparado un helado. **Tip:** resolverlo utilizando una lambda.

```
maquinasQuePreparanBien helado maquinas = ...
```

No se puede utilizar recursividad en ningún paso de este punto.

Casos de prueba que debe definir

Condición	Qué se espera
Cuántas máquinas preparan bien el helado de "frutilla al agua" a -5°C con ingredientes "frutilla" y "agua" (el mismo del punto 1)	4
Cuántas máquinas preparan bien el helado de "chocolate intenso" a -2°C (el mal preparado del punto 1)	1

2.4.3 c. BONUS Ingrediente favorito : común

Saber cuál es el ingrediente que más veces aparece dada una lista de helados.

Consejos de viejo haskellero:

1. primero obtener todos los ingredientes de los helados, sin repetidos
2. Luego saber cuántas veces aparece un ingrediente en un helado (podrías armar cof cof... una tup... cof cof, qué tos!)
3. Sacar el máximo de eso

No se puede utilizar recursividad en ningún paso de este punto.

Casos de prueba a definir

Dada esta lista de helados

```
helados = [  
    Helado "pistacho" 2 ["pistacho", "leche", "agua"],  
    Helado "frutilla" (-2) ["frutilla", "leche"],  
    Helado "dulce de leche" (-1) ["dulce de leche", "agua"],  
    Helado "agua" 0 ["agua"]  
]
```

El ingrediente más elegido debe ser “agua”.