

Gnomos!!!

Queremos modelar parte de un juego inspirado en el Gnomoria, donde se maneja una aldea y la misma puede transformarse con el trabajo de los gnomitos que la habitan. Para ello definimos modelamos los materiales de construcción para los edificios, los edificios para construir en la aldea y la aldea misma de la siguiente forma:

```
data Material = Material {  
    nombre :: String,  
    calidad :: Int  
} deriving (Show, Eq)
```

```
data Edificio = Edificio {  
    tipoEdificio :: String,  
    materiales :: [Material]  
} deriving (Show, Eq)
```

```
data Aldea = Aldea {  
    poblacion :: Int,  
    materialesDisponibles :: [Material],  
    edificios :: [Edificio]  
} deriving (Show, Eq)
```

Maximizando el uso de orden superior, aplicación parcial y composición, se requiere:

1. Desarrollar las siguientes funciones:

a. **esValioso** que recibe un material y retorna true si su calidad es mayor a 20

```
> esValioso (Material "Madera de pino" 25)  
True
```

b. **unidadesDisponibles** que recibe el nombre de un material y una aldea y retorna la cantidad de materiales disponibles con ese nombre en la aldea

```
> disponibles "Acero" (Aldea 50 [(Material "Acero" 15), (Material "Acero" 20),  
    (Material "Piedra" 5)] [])  
2
```

c. **valorTotal** recibe una aldea y retorna la suma de la calidad de todos los materiales que hay en la aldea. Estos son tanto los disponibles como los usados en sus edificios.

```
> valorTotal (Aldea 50 [(Material "Acero" 15), (Material "Piedra" 5)] [(Edificio  
    "Barracas" [(Material "Acero" 20)])])  
40
```

2. Desarrollar las siguientes tareas para que los gnomos puedan realizar en una aldea:

a. **tenerGnomito** que aumenta la población de la aldea en 1.

b. **lustrarMaderas** que aumenta en 5 la calidad de todos los materiales disponibles cuyo nombre empiece con la palabra "Madera". El resto de los materiales disponibles de la aldea no deberían verse afectados al realizar esta tarea.

c. **recolectar** que dado un material y una cantidad de cuánto de ese material se quiere recolectar,

incorpore a los materiales disponibles de la aldea ese mismo material tantas veces como se indique.

3. Realizar las consultas que permitan:

- a. Obtener los edificios chetos de una aldea, que son aquellos que tienen algún material valioso.
- b. Obtener una lista de nombres de materiales comunes, que son aquellos que se encuentran en todos los edificios de la aldea.

4.

- a. Definir la función **realizarLasQueCumplan::[Tarea]->(Aldea->Bool)->Aldea->Aldea**, que recibe una lista de tareas, un criterio que debería cumplir la aldea luego de realizar cada tarea y la aldea inicial, y retorne cómo quedaría la aldea si se realizaran las tareas válidas, una tras otra. Una tarea es válida si, después de realizarse sobre una aldea (la original o la resultante de haber realizado otra tarea previa), la misma cumple con el criterio indicado.
- b. Hacer consultas utilizando **realizarLasQueCumplan** de forma tal que:
 - i. Se tengan gnomitos 3 veces (como 3 tareas independiente entre sí), asegurando que siempre haya más unidades de comida disponible que la cantidad de población en la aldea luego de realizar cada tarea de tener gnomitos. Un material con el nombre "Comida" se considera una unidad de comida.
 - ii. Se recolecten 30 unidades de madera de pino de calidad igual a la calidad máxima de las maderas disponibles y luego se lustren las maderas disponibles de la aldea, asegurando siempre que todos los materiales disponibles sean valiosos.