



En todo jardín nos encontramos con una situación en común: el ataque de los zombies. Es nuestro deber defender nuestros hogares de estos comedores de cerebros.

La forma más efectiva de repeler el ataque de zombies, claro está, es por medio de plantas, como todos sabemos.

Para afrontar la grave situación en nuestros jardines se necesita de la codificación de un programa Haskell que modele esta problemática.



Nuestro querido vecino “Crazy Dave” nos cuenta que, para defendernos de los ataques zombies, se plantean líneas de defensa donde cada línea tiene una serie de plantas que protegen la casa de aquellos zombies que pasan por la misma; por ejemplo una línea podría tener dos SunFlowers, seguidas de una PeaShooter (es decir, 3 en total), o bien dos SunFlowers, una Nut y una PeaShooter (4 plantas), y a su vez tener los zombies que la atacan, como podrían ser 2 zombies básicos y uno con un cono en la cabeza. Más adelante se darán ejemplos más concretos.

#### **Se pide:**

1. Modelar las plantas y zombies definidos a continuación mediante funciones constantes:
  - a) Cada planta está definida por una especie, una cantidad de puntos de vida, una cantidad de soles que produce y un poder de ataque que daña a un zombie. Aquí varios ejemplos que se requieren modelar:

- PeaShooter: No genera soles, tiene 5 puntos de vida y su nivel de ataque es de 2 puntos.
- Repeater: Es igual al Peashooter, excepto que su nivel de ataque es el doble.
- Sunflower: Tiene 7 puntos de vida, produce 1 sol y no hace ningún tipo de daño.
- Nut: No hace daño ni genera soles... pero aguanta 100 mordiscos.
- Modelar también dos plantas nuevas con sus características particulares.

b) Los zombies cuentan con un nombre, algunos accesorios y un valor que representa el daño que le causan a una planta con su mordida:

- Zombie Base: Es el zombie básico, no lleva ningún accesorio y su mordida hace 1 punto de daño.
- Balloon Zombie: Es muy similar al zombie básico, con la diferencia de que cuenta con un globo entre sus accesorios.
- Newspaper Zombie: Como su nombre lo indica, lleva consigo un diario a todas partes, y hace 2 puntos de daño con su mordida.
- Gargantuar: Su nombre completo es 'Gargantuar Hulk Smash Puny God'. Carga con un poste eléctrico que encontró (quién sabe dónde) y lleva también un zombie enano, que posiblemente sea su mejor amigo. Muerde con una fuerza de 30 puntos.



Además, para todo zombie, se debe poder determinar su **nivel de muerte**. El mismo equivale a la cantidad de letras que tiene su nombre.

2. Se requiere poder:

a. Determinar de qué especialidad es una planta. Tener en cuenta para esto que:

- Las plantas que dan soles son de especialidad "Provedora"
- Las plantas cuyo poder de ataque es mayor a su nivel de vida, son de especialidad "Atacante".
- Si no cumplen con ninguna de las condiciones anteriores son de especialidad "Defensiva".

Por ejemplo:

```
> especialidad sunflower
"Proveedora"
```

- b. Determinar si un zombie es peligroso. Esto ocurre cuando tiene más de un accesorio o su nivel de muerte es mayor a 10.

Por ejemplo:

```
> esPeligroso gargantuar
True
```

```
> esPeligroso zombieBase
False
```

3. Como nos decía Crazy Dave anteriormente, los jardines están formados por líneas de defensa, en las cuales puede haber tanto plantas como zombies (que están atacando por esa línea). Algunos ejemplos de líneas son:

```
linea1 = LineaDeDefensa {
    plantas = [sunflower, sunflower, sunflower],
    zombies = []
}
linea2 = LineaDeDefensa {
    plantas = [peaShooter, peaShooter, sunflower, nut],
    zombies = [basicZombie, newspaperZombie]
}
linea3 = LineaDeDefensa {
    plantas = [sunflower, peaShooter],
    zombies = [gargantuar, basicZombie, basicZombie]
}
```

Se pide:

- a. Crear las funciones que hagan falta para:

- Agregar una planta a una línea (se agrega al final).
- Agregar un zombie a una línea (se agrega al final).

**Nota:** ¡No repetir código!

- b. Saber si una línea está en peligro, que es cuando el total de ataque de todas las plantas es inferior al total de mordiscos de todos los zombies, o bien todos los zombies de esa línea son peligrosos y hay al menos un zombie.

```
> estaEnPeligro linea2
False
```

```
> estaEnPeligro linea3
True
```

- c. Poder determinar si una línea necesita ser defendida, esto pasa cuando todas las plantas de esa línea son proveedoras. Por ejemplo:

```
> necesitaSerDefendida linea1
True
```

```
> necesitaSerDefendida linea3
```

False

4. Saber si una línea es mixta, que es cuando ninguna de las plantas de la línea tiene la misma especialidad que su inmediata siguiente. Además, la línea debe tener al menos dos plantas.

**Nota:** No usar length (ni ninguna función que tenga el mismo propósito).

```
> lineaMixta linea2  
False
```

5. Conocer el resultado del ataque de...

- a. Una planta a un zombie: Cuando una planta ataca a un zombie, lo daña según su potencia de ataque, pero ese daño consiste en quitarle al nombre del zombie las primeras n letras, siendo n la potencia de ataque de la planta. Cuando el nivel de muerte del zombie llega a cero, es derrotado, pero eso no se refleja acá.
- b. Un zombie a una planta: El zombie muerde a la planta causándole un daño a la vida equivalente a la fuerza de su mordida.