

1. En nuestra aplicación manejaremos Seres, que se identifican por un id, y tienen una posición (en 2D) y un indicador de movilidad.

- El id se compone de *nombrede la clase+secuencial*. El secuencial no se puede repetir en dos seres. No cambiará durante toda la vida del ser.
- La posición viene dada por un punto. Sí puede cambiar, desde dentro de la propia clase (ver siguiente párrafo)
- El indicador de si es o no móvil (verdadero/falso) se fija al crear el ser en concreto y ya no se podrá cambiar.

A cualquier ser se le puede pedir que se mueva a un punto diferente:

- Si el ser es móvil, informará con un mensaje (`Humano2: desplazándome a (0,0)`) y cambiará su posición.
- Si no, avisará con un mensaje (`Planta5: ¡carezco de movilidad!`), y no hará nada.

No hay seres genéricos. Deben ser forzosamente de algún subtipo. De momento hay dos subtipos (pero puede haber más en el futuro):

2. Biológico. Tienen una vida (inicialmente 100, su máximo).

Todos los seres biológicos pueden:

- Alimentar. Reponen la vida al máximo. Cada subtipo lanza un mensaje diferente que siempre comienza con el id.
- Sufrir una cierta cantidad. Pierde esa cantidad de vida (sin llegar a ponerse negativa). TODOS los seres biológicos que pueda haber lo harán así y no hay posibilidad de modificar ese comportamiento.

2.0. No hay seres biológicos genéricos. Deben ser forzosamente de algún subtipo. De momento hay tres subtipos (pero puede haber más en el futuro). Ninguno de los tres admite subtipos:

2.1. Humano. Es móvil. Al alimentarse dice “¡Está bueno esto!”. Para crearlo hay que proporcionar un punto y un nombre. Si no se proporciona punto se elegirá uno al azar de entre las ubicaciones predefinidas (ver sección ubicaciones).

2.2. Animal. Es móvil. Al alimentarse dice “¡Ñam, ñam!”. Para crearlo hay que proporcionar un punto y un alcance. Si no se proporciona punto se elegirá uno al azar de entre las ubicaciones predefinidas (ver sección ubicaciones).

Los animales pueden detectar: muestran un listado de todos los seres que estén a una distancia menor o igual que su alcance, ordenados de más cercanos a más lejanos.

2.3. Planta. No es móvil. Al alimentarse dice “¡Savia va, savia viene!”. Para crearlo hay que proporcionar un punto.

3. Mecánico. Tienen un indicador de encendido. Cuando se crean siempre están apagados.

Todos los seres mecánicos actuales o futuros pueden encenderse y apagarse, y lo harán forzosamente todos de la misma manera: mostrando mensajes como “Robot7: ...ON...” o “Robot7: ...OFF...”

3.0. No hay seres mecánicos genéricos. Deben ser forzosamente de algún subtipo. De momento hay dos subtipos (pero puede haber más en el futuro):

3.1. Robot. Es móvil. Para crearlo hay que proporcionar un punto.

3.2. Herramienta. No es móvil. Para crearlo hay que proporcionar un punto.

## COMPORTAMIENTOS

### 4. Manipulable.

Conlleva la capacidad serManipulado. No tiene una manera por defecto. Cada uno es manipulado a su manera

De momento, el único manipulable es Herramienta: mostrará un mensaje como “Herramienta2: BROM! PLAS! CRANCK!” o “...Herramienta2...está apagado”, dependiendo de si está encendida o apagada respectivamente.

### 5. Inteligencia. Presente (de momento) en humanos y robots.

Conlleva las capacidades

5.1. Manipular. Obviamente, sólo se puede manipular algo que sea manipulable. En principio, todo el que tenga inteligencia manipulará de la misma manera: invocando la acción serManipulado del objeto manipulable.

5.2. Aprender. En principio, todo el que tenga inteligencia aprende de la misma manera: mostrando un mensaje con la información del sujeto seguida de “...aprendiendo algo nuevo”.

En ambos casos, los robots tendrán el comportamiento por defecto si están encendidos. En caso contrario, mostrarán un mensaje como “...Robot1...está apagado”

## UBICACIONES

Son etiquetas o denominaciones asociadas a puntos concretos especiales. Son invariables.

Esta es la lista de denominaciones y los puntos que les corresponden:

ORIGEN → (0,0)

ESTE → (10,0)

OESTE → (-10,0)

NORTE → (0,10)

SUR → (0,-10)

Debe implementar la siguiente funcionalidad: devolver el punto asociado a una de las ubicaciones al azar.