

## Armadura Iron Man:

J.A.R.V.I.S. es una inteligencia artificial desarrollada por Tony Stark. Está programado para hablar con voz masculina y acento británico. Actualmente se encarga de todo lo relacionado con la información doméstica de su casa, desde los sistemas de calefacción y refrigeración hasta los Hot Rod que Stark tiene en su garage.

Tony Stark quiere adaptar a J.A.R.V.I.S. para que lo asista en el uso de sus armaduras, por lo tanto, serás el responsable de llevar adelante algunas de estas tareas.

El objetivo de JARVIS es que analice intensivamente toda la información de la armadura y del entorno y en base a esto tome decisiones inteligentes.

En este trabajo se deberá crear en el proyecto una clase llamada Armadura que modele toda la información y las acciones que pueden efectuarse con la Armadura de Iron Man.

La armadura de Iron Man es un exoesqueleto mecánico ficticio usado por Tony Stark cuando asume la identidad de Iron Man. La primera armadura fue creada por Stark y Ho Yinsen, mientras estuvieron prisioneros.

Las armaduras de Stark se encuentran definidas por un color primario y un color secundario. Se encuentran compuestas de dos propulsores, uno en cada bota; y dos repulsores, uno en cada guante (los repulsores se utilizan también como armas). Tony los utiliza en su conjunto para volar.

La armadura tiene un nivel de resistencia, que depende del material con el que está fabricada, y se mide con un número entero cuya unidad de medida de dureza es Rockwell ([https://es.wikipedia.org/wiki/Dureza\\_Rockwell](https://es.wikipedia.org/wiki/Dureza_Rockwell)). Todas las armaduras poseen un nivel de salud el cual se mide de 0 a 100. Además, Tony tiene un generador, el cual le sirve para salvarle la vida en cada instante de tiempo alejando las metrallas de metal de su corazón y también para alimentar de energía a la armadura. La batería a pesar de estar en el pecho de Tony, es considerada como parte de la armadura.

La armadura también posee una consola en el casco, a través de la cual JARVIS le escribe información a Iron Man. En el casco también se encuentra un sintetizador por donde JARVIS susurra cosas al oído de Tony. Cada dispositivo de la armadura de

Iron Man (botas, guantes, consola y sintetizador) tienen un consumo de energía asociado.

En esta primera etapa con una armadura podremos: caminar, correr, propulsar, volar, escribir y leer.

- Al caminar la armadura hará un uso básico de las botas y se consumirá la energía establecida como consumo en la bota por el tiempo en el que se camine.
- Al correr la armadura hará un uso normal de las botas y se consumirá el doble de la energía establecida como consumo en la bota por el tiempo en el que se corra.
- Al propulsarse la armadura hará un uso intensivo de las botas utilizando el triple de la energía por el tiempo que dure la propulsión.
- Al volar la armadura hará un uso intensivo de las botas y de los guantes un uso normal consumiendo el triple de la energía establecida para las botas y el doble para los guantes.
- Al utilizar los guantes como armas el consumo se triplica durante el tiempo del disparo.
- Al utilizar las botas para caminar o correr el consumo es normal durante el tiempo que se camina o se corra.
- Cada vez que se escribe en la consola o se habla a través del sintetizador se consume lo establecido en estos dispositivos. Solo se usa en nivel básico.
- Cada vez que se efectúa una acción se llama a los métodos usar del dispositivo se le pasa el nivel de intensidad y el tiempo. El dispositivo debe retornar la energía consumida y la armadura deberá informar al generador se ha consumido esa cantidad de energía.

Modele las clases, los atributos y los métodos necesarios para poder obtener un modelo real de la armadura y del estado de la misma.

### Mostrando Estado

Hacer un método que JARVIS muestre el estado de todos los dispositivos y toda la información de la Armadura.

### Estado de la Batería

Hacer un método para que JARVIS informe el estado de la batería en porcentaje a través de la consola. Poner como carga máxima del reactor el mayor float posible. Ejecutar varias acciones y mostrar el estado de la misma.

### Mostrar Información del Reactor

Hacer un método para que JARVIS informe el estado del reactor en otras dos unidades de medida. Hay veces en las que Tony tiene pretensiones extrañas. Buscar en Wikipedia la tabla de transformaciones.

### Sufriendo Daños

A veces los dispositivos de la armadura sufren daños para esto cada dispositivo contiene un atributo público que dice si el dispositivo se encuentra dañado o no. Al utilizar un dispositivo existe un 30% de posibilidades de que se dañe.

La armadura solo podrá utilizar dispositivos que no se encuentren dañados.

Modifique las clases que sean necesarias para llevar adelante este comportamiento.

### Reparando Daños

Hay veces que se puede reparar los daños de un dispositivo, en general es el 40% de las veces que se puede hacer. Utilizar la clase Random para modelar este comportamiento.

En caso de estar dentro de la probabilidad (es decir probabilidad menor o igual al 40%) marcar el dispositivo como sano. Si no dejarlo dañado.

### Revisando Dispositivos

Los dispositivos son revisados por JARVIS para ver si se encuentran dañados. En caso de encontrar un dispositivo dañado se debe intentar arreglarlo de manera insistente. Para esos intentos hay un 30% de posibilidades de que el dispositivo quede destruido, pero se deberá intentar arreglarlo hasta que lo repare, o bien hasta que quede destruido.

Hacer un método llamado revisar dispositivos que efectúe lo anteriormente descrito, el mecanismo insistente debe efectuarlo con un bucle do while.

### Radar Versión 1.0

JARVIS posee también incorporado un sistema que usa ondas electromagnéticas para medir distancias, altitudes, ubicaciones de objetos estáticos o móviles como aeronaves, barcos, vehículos motorizados, formaciones meteorológicas y por su puesto enemigos de otro planeta.

Su funcionamiento se basa en emitir un impulso de radio, que se refleja en el objetivo y se recibe típicamente en la misma posición del emisor.

Las ubicaciones de los objetos están dadas por las coordenadas X, Y y Z. Los objetos pueden ser marcados o no como hostiles. JARVIS también puede detectar la resistencia del objeto, y puede detectar hasta 10 objetos de manera simultánea.

JARVIS puede calcular la distancia a la que se encuentra cada uno de los objetos, para esto siempre considera que la armadura se encuentra situada en la coordenada (0,0,0).

Hacer un método que informen a qué distancia se encuentra cada uno de los enemigos.

Usar la clase Math de Java.

### Simulador

Hacer un método en JARVIS que agregue en radar objetos, hacer que la resistencia, las coordenadas y la hostilidad sean aleatorios utilizando la clase random. Utilizar la clase Random.

¿Qué ocurre si quiero añadir más de 10 objetos?

¿Qué ocurre si cuando llevo 8 enemigos aumento la capacidad del vector?

### Destruyendo Enemigos

Desarrollar un método para que JARVIS que analice todos los objetos del radar y si son hostiles que les dispare. El alcance de los guantes es de 5000 metros, si el objeto se encuentra fuera de ese rango no dispara.

JARVIS al detectar un enemigo lo atacará hasta destruirlo, la potencia del disparo es inversamente proporcional a la distancia a la que se encuentra el enemigo y se descontará de la resistencia del enemigo. El enemigo se considera destruido si su resistencia es menor o igual a cero.

JARVIS solo podrá disparar si el dispositivo está sano y si su nivel de batería lo permite.

Si tiene los dos guantes sanos podrá disparar con ambos guantes haciendo más daño.  
Resolver utilizando un for para recorrer el arreglo y un while para destruir al enemigo.

### Acciones Evasivas

Desarrollamos un método para que JARVIS que analice todos los objetos del radar y si son hostiles que les dispare. Modificar ese método para que si el nivel de batería es menor al 10% se corten los ataques y se vuelve lo suficientemente lejos para que el enemigo no nos ataque. Deberíamos alejarnos por lo menos 10 km enemigo. Tener en cuenta que la velocidad de vuelo promedio es de 300 km / hora.