

## Instrucciones

Se te proporciona un guión con la mecánica de juego y un proyecto NetBeans con el código del juego, obviamente, incompleto<sup>1</sup>. En el siguiente apartado, se te especifican las tareas que debes realizar para completarlo.

NOTA: verás infinidad de “errores” en el código destacados por el IDE. No te preocupes por ellos. Si realizas las tareas que se te piden correctamente, todos ellos desaparecerán. Modifica / añade EXCLUSIVAMENTE lo que se te pide. Si empiezas a modificar el resto del código, te harás un lío enorme.

RA involucrados en este examen: 4-Desarrolla programas organizados en clases analizando y aplicando los principios de la programación orientada a objetos. 7-Desarrolla programas aplicando características avanzadas de los lenguajes orientados a objetos y del entorno de programación.

## Tareas

1. (2p) Crea un archivo con un determinado elemento del lenguaje Java (llámalo “Arma”) para poder gestionar los datos de las armas correctamente. Cada arma debe tener exactamente los datos y valores reflejados en la tabla que se proporciona en la sección “Armas”.

Modifica la línea 20 del archivo “Guerrero.Java” para que se asocie al campo “arma” una de las armas al azar. TEN EN CUENTA que, en el futuro, el archivo “Arma.java” al que nos referimos en el párrafo anterior podría contener datos de más armas (ej: látigo).

2. (1p) Modifica las líneas 12, 21, 22 y 23 del archivo “Personaje.java” para que se cumpla con lo especificado con respecto a “Personaje” en la sección “Jerarquía”.

3. (1p) Modifica las líneas 14 y 22 del archivo “Hroe.Java” para que se cumpla con lo especificado con respecto a “Hroe” en la sección “Jerarquía”.

4. (1p) Modifica la línea 12 del archivo “Villano.Java” para que se cumpla con lo especificado con respecto a “Villano” en la sección “Jerarquía”.

Modifica las declaraciones (una línea en cada uno) de los métodos “generaVillanoaleatorio”, “generaNombreAleatorio”, “vocalAleatoria” y “consonanteAleatoria” para que sean coherentes con su contenido y con el comentario que los precede.

5. (1p) Modifica la línea 12 del archivo “Guerrero.Java” y la línea 12 del archivo “Nigromante.Java” para que se cumpla con lo especificado con respecto a “Guerrro” y “Nigromante” en la sección “Jerarquía”.

6. (2p) Modifica las líneas 12, 18 y 29 de “Defensa.java”, la línea 12 de “Guerrero.java” y la línea 14 de “Hroe. Java” para que se cumpla con lo especificado con respecto a “capacidad de Defensa” en la sección “Jerarquía”.

7. (2p) En “Villano.java” , realiza las acciones necesarias para dotar a los villanos de un “orden natural”: primero los de menor energía y, en caso de empate, primero los que llevan más tiempo en el juego.

En el archivo “Principal.java”, fata la última línea del método “quitaVillanosMuertos”.

Escríbela tú: debe ordenar “de forma natural” el array *villanos*, dejando los valores nulos (si los hubiere) al final del mismo.

---

<sup>1</sup>Todo el código está escrito con elementos del lenguaje que hemos visto en clase; así qué, si quieres echar un vistazo a alguna de las partes en detalle, deberías entenderla sin problemas.