

Docente: Ing. Jannelly Bello
Cátedra: Algoritmos y Programación III

Laboratorio II

Implementación POO Básico + UML

Entre los años 1100 y 1300 a.c, París de Troya secuestró a Helena de Grecia, la mujer más bella del mundo. Melenao de Esparta, su esposo, convocó a los griegos a que emprendieran una guerra contra los troyanos. Se pide crear un programa que represente los elementos de esta guerra, incluyendo el caballo de Troya.

Los guerreros pueden ser troyanos o griegos. De los guerreros se desea conocer su nombre, edad, fuerza y si está herido o muerto. La edad debe estar entre 5 y 60 (ambos inclusive) y la fuerza entre 1 y 10 (ambos inclusive). El programa debe permitir comprobar la edad y la fuerza, indicando si están correctos o no.

Al crear un guerrero, no puede estar herido ni muerto. Además, se debe comprobar que la fuerza y la edad sean válidas, de lo contrario se coloca por defecto la edad en 25 y la fuerza de 5. También se pueden crear guerreros con los siguientes valores por defecto: nombre GuerreroX, edad 15 y fuerza 1. También, se pueden crear copias de guerreros, en cuyo caso solo se le cambia el nombre antiguo por el nuevo nombre.

Los griegos pueden retirarse de la batalla siempre y cuando no estén heridos ni evidentemente muertos. En cambio, los troyanos no pueden retirarse, sin importar si están heridos. En cada caso, el programa de informar si puede retirarse o no y el porqué.

El caballo de Troya se caracteriza por tener capacidad que representa la cantidad de guerreros griegos que pueden estar dentro del caballo, ocupación que representa la cantidad actual de guerreros que se encuentran en el caballo y los ocupantes que muestran el detalle de los guerreros griegos que están en el caballo.

Al montarse un guerrero en el caballo se debe verificar que sea griego, de lo contrario no se puede montar devolviendo -2. Si se monta, debe devolver el número de ocupantes actuales. Si el caballo está lleno se debe informar que no se pueden montar más guerreros y devolver -1. También los guerreros se pueden bajar del caballo, por lo que se debe actualizar la ocupación del mismo.

Además, se debe permitir buscar guerreros en el caballo por su nombre, si está devuelve la posición donde se encuentra de lo contrario devuelve -1.

Instrucciones:

Se pide:

1. Realizar el diagrama de clases estableciendo la asociación, cardinalidad y relaciones existentes entre las clases.
2. Implementación en JAVA del modelo planteado. (Definición de atributos, constructores con y sin parámetros, métodos get/set, otros métodos según se requiera, etc.).
3. En la solución del ejercicio se debe reflejar: Sobreescritura de métodos, sobrecarga de métodos, uso de la sentencia super, uso de la referencia this, clase abstracta.
4. En el Main mostrar la información de los guerreros, montar, desmontar guerreros guerreros reflejando el polimorfismo. Además dejar plasmado la ejecución del resto de las acciones de guerreros y caballo.