

Proyecto II

Mundo Chiquito

(20 pts)

Introducción

En el popular *JCC* (Juego de Cartas Coleccionables) **Duelo de cartas de mostro** los jugadores (llamados duelistas) construyen mazos de entre 40 y 60 cartas para competir entre sí. En este juego existen dos clases de carta, las cartas **mostro** y las cartas **conjuro**.

Las cartas **mostro** representan a criaturas que combaten en nombre de su duelista. Las cartas **mostro** tienen distintas características que las distinguen. Cada **mostro** tiene un **nombre** único, un **nivel** representado por un número entero entre 1 y 12, una magnitud de **poder** representada por un número entero múltiplo de 50 y un **atributo** que puede ser cualquiera entre **AGUA**, **FUEGO**, **VIENTO**, **TIERRA**, **LUZ**, **OSCURIDAD** o **DIVINO**.

Las cartas **conjuro** son más sencillas, cada una contiene instrucciones que el jugador de turno que la utiliza debe aplicar. Existe una carta **conjuro** muy particular llamada **Mundo Chiquito** cuyo texto es el siguiente:

Mundo Chiquito

Muéstrale a tu oponente una carta mostro de tu mano, elige un mostro de tu mazo que comparta exactamente una característica (nivel, poder o atributo) con el mostro revelado de tu mano y muéstraselo a tu oponente. Luego agrega desde tu mazo a tu mano un mostro que comparta exactamente una característica (nivel, poder o atributo) con el mostro revelado de tu mazo.

Mundo Chiquito es una carta poderosa con el potencial de revolucionar la manera en que los duelistas construyen sus mazos, dado que le da al jugador la facilidad de buscar cartas específicas sin tener que depender de la suerte para robarlas. Pero sus condiciones son tan complicadas que es difícil para los jugadores saber que cartas les pueden servir de ‘puente’ entre el **mostro** que tienen en su mano y el **mostro** que quieren buscar de su mazo.

Es por ello que los jugadores competitivos de **Duelo de cartas de mostro** acudieron a los estudiantes de la **Universidad Simón Bolívar** para desarrollar un algoritmo que les ayude a encontrar todas las posibles combinaciones de cartas **mostro** que satisfacen las condiciones de **Mundo chiquito**.

Planteamiento del problema

Se desea que implemente una clase **CartaMostro** en un archivo **CartaMostro.java** con un atributo privado por cada característica de una carta **mostro** en **Duelo de cartas de mostro** (nombre, nivel, poder, atributo). Debe tener un constructor que asigne todos estos atributos (validando que cumplan con las restricciones) y un método **getter** para cada uno. Por ejemplo, un método **getNombre** que retorne el nombre de la instancia.

Se desea que implemente la solución al problema descrito en una clase **MundoChiquito** en un archivo **MundoChiquito.java**. El método **main** de esta clase debe leer de un archivo **deck.csv** que contiene los datos de un mazo de **Duelo de cartas de mostro** (excluyendo las cartas conjuro, que son irrelevantes para el problema) en el siguiente formato:

```
deck.csv
Nombre,Nivel,Atributo,Poder
mostroUno,5,LUZ,2100
mostroDos,7,LUZ,2400
mostroTres,7,VIENTO,2500
```

Se debe leer cada línea del archivo y crear una instancia de **CartaMostro** por cada línea (con la excepción del header). Estas instancias deben ser usadas para construir un grafo cuyos vértices sean de tipo **CartaMostro**. La implementación del grafo se deja a su criterio.

Con el grafo construido, se debe hallar todas las **ternas** de **CartaMostro** que satisfacen las condiciones de **MundoChiquito** tales como se describen en el texto de la carta. En estas ternas, la primera carta es la que se revela de la mano, la segunda es la que se revela del mazo y la última es la que se agrega del mazo a la mano.

Finalmente, se deben imprimir por la salida estándar los nombres de las cartas que forman parte de cada terna separados por espacios (una terna por línea). En el caso del archivo **deck.csv** dado como ejemplo, las ternas serían.

```
mostroUno mostroDos mostroTres
mostroTres mostroDos mostroUno
mostroUno mostroDos mostroUno
mostroDos mostroUno mostroDos
mostroDos mostroTres mostroDos
mostroTres mostroDos mostroTres
```

Nótese que hay ternas que comienzan y terminan con la misma carta. Esto se debe a que un jugador puede incluir múltiples copias de la misma carta en su mazo. El archivo **deck.csv** no debe contener copias repetidas, pero la posibilidad de copias repetidas debe ser tomada en cuenta para la solución.

Entrega

Este proyecto se realizará en equipos de 2 integrantes y debe ser subido a *GitHub*. Su repositorio debe contener las clases descritas y un archivo **README.md** identificado con el nombre y número de carnet de los integrantes junto con una breve explicación de su implementación. La fecha límite de entrega es el **miércoles 15 de noviembre de 2023 a las 11:59 pm**.