

Indicaciones Generales

- Los grupos deberán tener exactamente tres (3) integrantes. En caso de que la cantidad total de estudiantes no sea múltiplo de 3, se admitirá excepcionalmente algún grupo de dos (2) integrantes.
- La fecha de entrega es el **Lunes 03 de diciembre de 2018** hasta las 22:00 hs. La entrega será realizada **por mail** a la dirección de correo indicada por el docente. Se debe entregar:
 - Un **documento en formato PDF** conteniendo:
 - La elección del diseño para el obligatorio, debidamente justificado, junto con el correspondiente desglose de los requerimientos.
 - La correspondiente especificación completa en UML para el diseño elegido.
 - Justificación de las estructuras de datos elegidas para la implementación de las colecciones que se utilicen. También se debe indicar en qué casos se trabajará con objetos y en qué casos con punteros a objetos.
 - Un **archivo .zip** conteniendo el **project** en Code::Blocks, incluyendo en él todos los archivos (.h) y (.cpp) correspondientes a las clases implementadas y al programa main. Se deben eliminar los archivos compilados (.o) de la carpeta obj y el archivo ejecutable (.exe) de la carpeta bin.
- Ambas partes de la entrega deberán estar debidamente identificadas con los nombres y las cédulas de todos los integrantes del grupo.

Objetivo

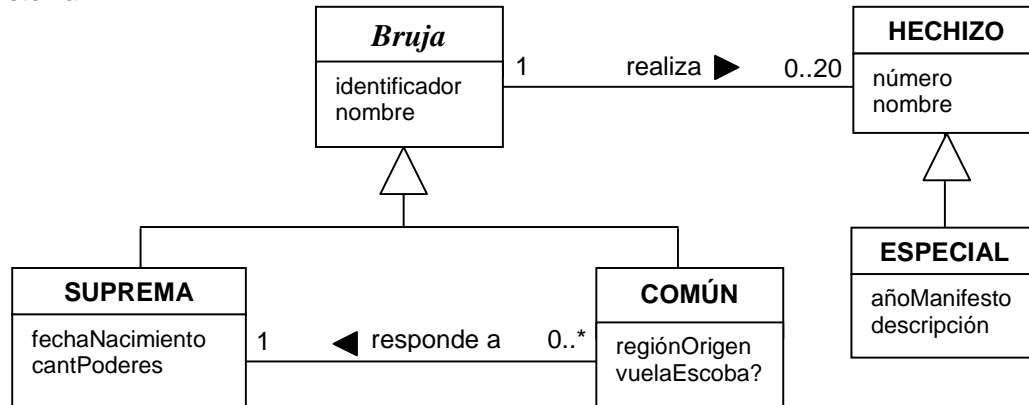
- Puesta en práctica de todos los conocimientos adquiridos durante el curso.

Planteo del Problema

El aquelarre de brujas **C.O.V.E.N** (Comunidad occidental de virtuosas empoderadas y nocturnas) necesita un programa para registrar los datos de las brujas que lo integran y de los hechizos que realizan. Sus integrantes descienden directamente de los linajes originales de Salem y, tras varios siglos desde la feliz abolición de las cacerías de brujas, han ido en aumento, así como el volumen de información correspondiente, por lo cual el aquelarre requiere de una pronta informatización.



Se desea implementar un pequeño prototipo en C++ que gestione únicamente la información de las brujas y sus hechizos. Por tratarse de una primera versión, no se registrarán otros datos por el momento. Se propuso el siguiente Diagrama de Clases Conceptual en UML para el análisis del sistema:



Hay dos clases de brujas, las supremas y las brujas comunes. Las supremas son las brujas más poderosas y a ellas responden las brujas comunes. De todas ellas se registra su identificador alfanumérico (único dentro del aquelarre) y su nombre. De las brujas supremas se registra además su fecha de nacimiento y su cantidad de poderes, mientras que de las brujas comunes se registra su región de origen y si vuelan o no en escoba (práctica que ha ido entrando progresivamente en desuso con los siglos, aunque aún quedan algunas que se aferran a las viejas tradiciones).

Cada bruja puede realizar hasta 20 hechizos, pues la astrología genética de los cromosomas que otorgan poderes a las brujas impide biológicamente un número mayor. De todos los hechizos se registra número y nombre. Algunos hechizos son especiales, en cuyo caso se registra además el año en que la bruja lo manifestó por primera vez, junto con una descripción de su magia. Interesa registrar los números de hechizo de manera consecutiva para cada bruja. El primer hechizo manifestado por una bruja tendrá el número 1, el siguiente el número 2 y así sucesivamente.

Cada bruja del aquelarre es ingresada al sistema inicialmente sin hechizos, cada uno de los cuales va siendo registrado en el sistema a medida que la bruja lo manifiesta exitosamente (esto es, cuando la bruja lo realiza por primera vez). Es importante evitar que, por error, se ingresen más de 20 hechizos para una misma bruja. También interesa que las brujas se registren ordenadas en forma alfabética por identificador. No existe cota para la cantidad de brujas en el aquelarre.

Una funcionalidad importante que el programa deberá brindar es el cálculo de los puntos de poder de cada bruja, los que se calculan de la siguiente manera.

- **Bruja suprema:** Se calcula sumando 2 puntos por cada hechizo común y 4 puntos por cada hechizo especial realizado por la bruja. A esto se suma 1 punto por cada poder manifestado por la bruja y 5 puntos adicionales si nació antes del comienzo del siglo 20.
- **Bruja común:** Se calcula sumando 1 punto por cada hechizo común y 2 puntos por cada hechizo especial realizado por la bruja. A esto se suma 1 punto adicional si nació en la región de Salem.

Requerimientos

Se desea que el Sistema resuelva los siguientes requerimientos:

- 1) Registrar una nueva bruja suprema que se incorpora al aquelarre.
- 2) Dados el identificador de una bruja suprema y los datos de una nueva bruja común que se incorpora al aquelarre, registrar la bruja común en el aquelarre, asignándole la suprema con dicho identificador.
- 3) Listado simple de todas las brujas registradas en el aquelarre, ordenado alfabéticamente por identificador de bruja. De cada bruja se listará su identificador, su nombre y su tipo de bruja (común o suprema). Los datos de una misma bruja se listarán en una misma línea.
- 4) Dado el identificador de una bruja, listar en detalle **todos** sus datos junto con sus puntos de poder. En caso de que sea una bruja común, listar también los datos de la bruja suprema a la cual responde junto con los puntos de poder de la suprema.
- 5) Listar los datos de la mayor bruja suprema registrada en el aquelarre. Es decir, la bruja suprema con la fecha de nacimiento más antigua de todas.
- 6) Dados el identificador de una bruja y los datos de un nuevo hechizo que realiza, registrar el nuevo hechizo para dicha bruja. El número del nuevo hechizo deberá ser consecutivo al número del último hechizo registrado hasta el momento para dicha bruja.
- 7) Dados el identificador de una bruja y un número de hechizo, listar todos los datos del hechizo correspondiente a ese número realizado por dicha bruja.
- 8) Dado el identificador de una bruja, contar cuántos hechizos comunes y cuántos hechizos especiales ha realizado dicha bruja.

Observaciones

- Al realizar el diseño, incorpore **todos** los métodos que sean necesarios a las clases (tanto primitivos como específicos). Haga **todos** los chequeos que sean necesarios para resolver cada requerimiento.
- Aplique el principio de **separación en capas** al definir la especificación de métodos. Debe incluir una clase **Fachada** de acceso a la capa lógica.
- Al implementar, debe respetar **estrictamente** el análisis y diseño realizados.
- En esta primera versión, los datos de brujas y hechizos **no** serán dados de baja del sistema.
- **Evite** duplicar información dentro del sistema. Para ello, defina apropiadamente en qué casos corresponde trabajar con **punteros** a objetos.
- Implemente las clases en forma incremental. Empiece por las clases más simples y siga luego con las clases más complejas, dejando la **Fachada** para el final. Haga un programa de prueba para cada clase que implemente.
- A efectos de simplificar el trabajo solicitado, en este obligatorio **no** se pide respaldo en disco de la información manejada por el sistema.