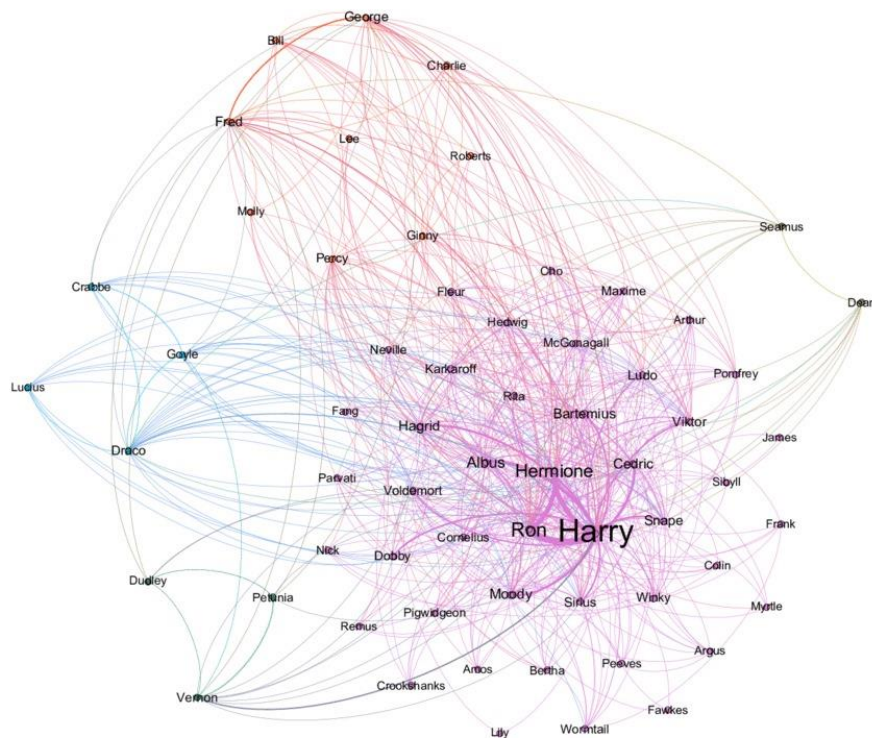


Trabajo práctico de laboratorio de estructuras de datos no lineales (nov/2023)

GRAFOS *Harry Potter*

Descripción general del problema¹

La empresa de desarrollo de videojuegos “Mega” nos ha propuesto formar parte de su división de desarrollo de juegos RPG (Role-play Game), comenzando por participar en el desarrollo de su nueva aventura gráfica ambientada en el Universo Harry Potter. Tenemos a nuestra disposición dos ficheros CSV con información que representan una red de relaciones entre personajes de Harry Potter, correspondientes a las interacciones que han tenido a lo largo de los 7 libros que componen la saga original. El fichero **characters.csv** contiene los datos de cada personaje (identificador, nombre, breve biografía, especie, género, casa, año de nacimiento, si es mago o no, herencia, color de ojos, color de pelo, tipo de varita y si sobrevive o no), y el fichero **relations.csv** contiene interacciones entre pares de personajes y si estas son positivas o negativas. Esta red puede visualizarse a través del siguiente grafo:



El objetivo de este trabajo práctico es implementar un programa en Java que procese los ficheros indicados y responda a los siguientes requisitos:

- Construir el grafo correspondiente y mostrar el número de personajes, el número total de relaciones entre personajes, el personaje que tiene relación con más personajes, el personaje que tiene relación con menos personajes y el personaje que tiene más relaciones positivas (y cuántas).

¹ Problema de grafos inspirado en el trabajo *Potter Network* (<https://github.com/efekarakus/potter-network>)

- b) Se necesita programar un equipo para una misión que se llamará “Young guns”, formado por magos jóvenes de carácter positivo. Para ello, dados dos nombres de personajes leídos por teclado (líderes del equipo), buscar una secuencia cualquiera de personajes positivos² que conecte a ambos líderes teniendo en cuenta que su especie deberá ser “human” o “half-giant”, su año de nacimiento mayor que 1975 y deberá ser mago. Nota: Los líderes no tienen por qué cumplir estas condiciones, se supone que siempre formarán parte de la selección.
- c) Se necesita programar otro equipo que se utilizará para las batallas de entrenamiento de los jugadores, generado dinámicamente a partir de dos personajes base. De esta forma, dados dos nombres de personajes leídos por teclado (líderes del equipo), indicar la secuencia más corta de personajes que conecte a ambos líderes teniendo en cuenta que estén vivos, pertenezcan a alguna casa, su herencia no sea “muggle” y no tengan el pelo negro (“black”). Nota: Los líderes no tienen por qué cumplir estas condiciones, se supone que siempre formarán parte de la selección.

Recomendaciones y requisitos

- En el apartado b) es obligatorio utilizar un algoritmo DFS.
- Se valorará que la búsqueda de caminos se repita a indicación del usuario.
- Se usará la implementación de Grafos proporcionada en la asignatura.
- Se utilizará el patrón *Elemento Decorado* visto en clase para decorar o recoger el conjunto de atributos técnicos relacionados con la manipulación y recorrido de grafos, el cual deberá basarse en tipos de datos genéricos.
- No se permitirá el uso de moldes, salvo para la implementación del método *equals* de la clase *Element*.
- Cada clase que se genere debe estar en un fichero diferente.
- El nombre de cada clase (y fichero) deberá comenzar por el código de grupo (ej., B2-07_...).
- Cada clase del programa deberá contener documentación interna. En Campus Virtual se puede encontrar información para generar esta documentación (sección Laboratorio).
- Con el fin de utilizar sistemas antiplagio y de detección semántica de similitudes, los ficheros de código fuente no podrán contener datos sobre los estudiantes de carácter personal como el nombre, el correo electrónico... En su lugar, se identificará a los autores y el grupo de prácticas en los comentarios del código con las iniciales de los estudiantes y el código de grupo asignado.
- No se podrán utilizar caracteres especiales en los nombres de los ficheros.
- Las clases y recursos se entregarán comprimidos en formato ZIP (ver Formato y entrega del trabajo), habiéndose organizado en una única carpeta raíz que no contenga subcarpetas.

Normas de realización y entrega

- El proyecto deberá realizarlo cada uno de los grupos de trabajo que se han formado previamente en clase de prácticas. No podrá realizarse de manera individual.
- La entrega y evaluación se realizará en el día que se indique previamente en el *Campus Virtual*. Uno de los miembros del grupo remitirá un fichero comprimido con todas las clases que componen el programa. De manera individual se deberá también remitir la plantilla de participación, que se encuentra disponible en el *Campus Virtual*.
- Para que la práctica sea evaluada, el programa deberá compilar y ejecutar correctamente.
- Se recuerda que esta actividad es de superación obligatoria al efecto de poder superar la asignatura.

² Se comprobará que se pase sólo por aristas que tengan valor “+”