

Práctica Individual 2

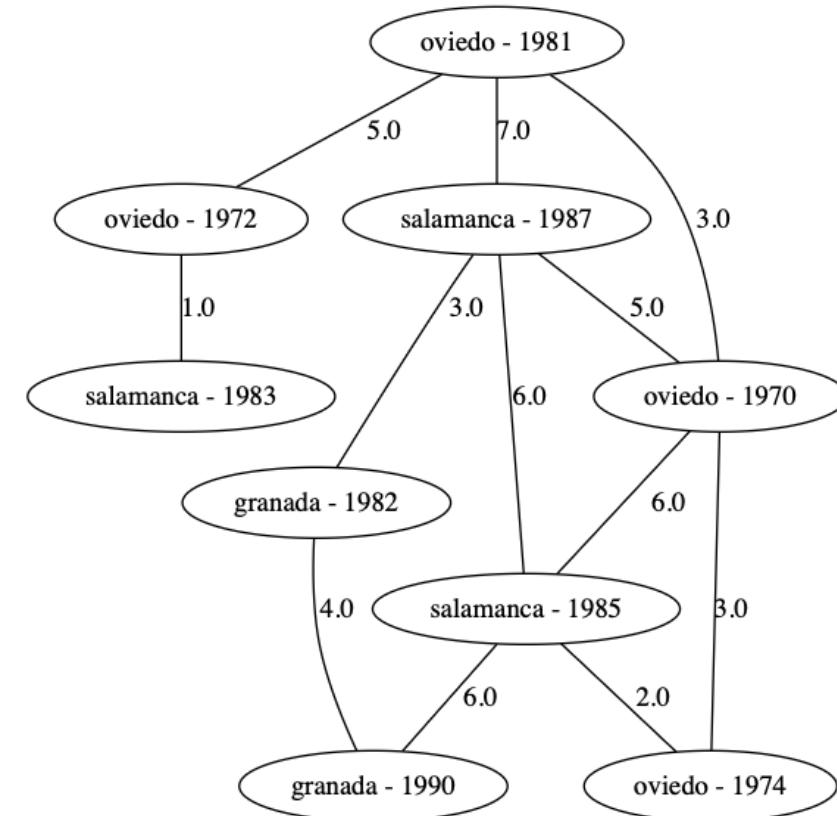
ESCUELA TÉCNICA SUPERIOR DE
INGENIERÍA INFORMÁTICA

Departamento de Lenguajes y Sistemas Informáticos
Curso 2025-2026

Belén Ramos Gutiérrez
brgutierrez@us.es

Ejercicio 3

Se tiene un conjunto de investigadores de prestigio que forman parte de una red, de los que se conoce su identificador numérico único, su año de nacimiento, y su universidad de origen. De cada par de investigadores se conoce el número de artículos que tienen en común (puede ser 0). Considere que el grafo que representa a los autores y sus colaboraciones es conexo.



Ejercicio 3a

- a) Obtenga una vista del grafo que sólo incluya:
- Los investigadores nacidos antes de 1982 o que tengan más de 5 artículos con alguno de sus coautores.
 - Las colaboraciones más relevantes (número de artículos mayor que 5).

Muestre el grafo configurando su apariencia de forma que se resalten los vértices y las aristas de la vista.

Ejercicio 3a

- a) Obtenga una **vista** del grafo que sólo incluya:
- Los investigadores nacidos antes de 1982 o que tengan más de 5 artículos con alguno de sus coautores.
 - Las colaboraciones más relevantes (número de artículos mayor que 5).

Muestre el grafo configurando su apariencia de forma que se resalten los vértices y las aristas de la vista.



toDot del grafo original

Ejercicio 3b

- b) Implemente un método que devuelva un conjunto con los 5 investigadores que tienen un mayor número de investigadores colaboradores.

Muestre el grafo configurando su apariencia de forma que se resalten dichos investigadores.



toDot del grafo original

Ejercicio 3b

-
- b) Implemente un método que devuelva un conjunto con los 5 investigadores que tienen un mayor número de investigadores colaboradores.

Muestre el grafo configurando su apariencia de forma que se resalten dichos investigadores.

Pistas:

- La existencia de una colaboración se modela con una arista
- ¿Cómo se llama la propiedad de un vértice que cuenta cuántas aristas están directamente conectadas a él? (`degreeOf`)
- Necesitaremos ordenar a todos los investigadores en función de la métrica de la Pista 2 y luego filtrar solo los que están en la cima. (`sort`, `comparator`, `limit`)

Ejercicio 3c

c) Implemente un método que devuelva un map que almacene para cada investigador una lista de investigadores colaboradores ordenados de mayor a menor número de artículos conjuntos con él.

Muestre el grafo configurando su apariencia de forma que se resalte para cada investigador su colaboración con mayor número de artículos.

toDot del grafo original

Ejercicio 3c

c) Implemente un método que devuelva un map que almacene para cada investigador una lista de investigadores colaboradores ordenados de mayor a menor número de artículos conjuntos con él.

Pistas:

- Necesitaremos el número de artículos conjuntos (el peso) entre el investigador A y el investigador B para poder ordenarlos. (`getEdgeWeight`)
- Cada arista proporciona un par de investigadores. Debemos usar estos pares para inicializar y poblar las listas de colaboradores para ambos.
- Las listas de colaboradores deben ordenarse:
 - Obtener el peso de las aristas I-C1 e I-C2 . Usar ordenación personalizada (`Comparator`) que acceda a estos pesos para tomar la decisión (Utiliza el resultado de la Pista 1).

Ejercicio 3d

d) La **distancia** entre 2 investigadores es el número mínimo de investigadores intermedios que los relacionan, o 0 en caso de que estén relacionados directamente entre sí. Determine cuál es el **par de investigadores** que están más alejados, o uno de los pares en caso de que haya más de uno.

Ejercicio 3d

d) La **distancia** entre 2 investigadores es el número mínimo de investigadores intermedios que los relacionan, o 0 en caso de que estén relacionados directamente entre sí. Determine cuál es el **par de investigadores** que están más alejados, o uno de los pares en caso de que haya más de uno.

Pistas:

- ¿Qué algoritmo de camino mínimo está diseñado para calcular la distancia mínima entre todos los pares de vértices ($V \times V$) de un grafo?(FloydWarshall)
- Necesitamos obtener la distancia más corta en número de saltos (aristas), ignorando el peso real (artículos en común) → `getPath(a,b).getLength()`
- El grafo es no dirigido , intentaremos calcular la distancia entre el mismo par de investigadores, una única vez.
- Una vez con todas las distancias, buscaremos el máximo (diámetro del grafo).

Ejercicio 3e

- e) Se desea fomentar nuevas colaboraciones entre investigadores que aún no han colaborado, para lo cual se van a organizar una serie de **reuniones** online (sin límite de asistentes).

Minimice el número de reuniones que deben organizarse de forma que **2 investigadores** colaboradores o que **tengan la misma universidad de origen** no pueden acudir a la misma reunión.

Si han colaborado o son de la misma universidad →

No pueden estar en la misma reunión →

Queremos el **MÍNIMO** de reuniones →

Y si fueran adyacentes...¿**COLOREADO**??

Ejercicio 3e

e) Se desea fomentar nuevas colaboraciones entre investigadores que aún no han colaborado, para lo cual se van a organizar una serie de **reuniones** online (sin límite de asistentes).

Minimice el número de reuniones que deben organizarse de forma que 2 investigadores colaboradores o que tengan la misma universidad de origen no pueden acudir a la misma reunión.

Pistas:

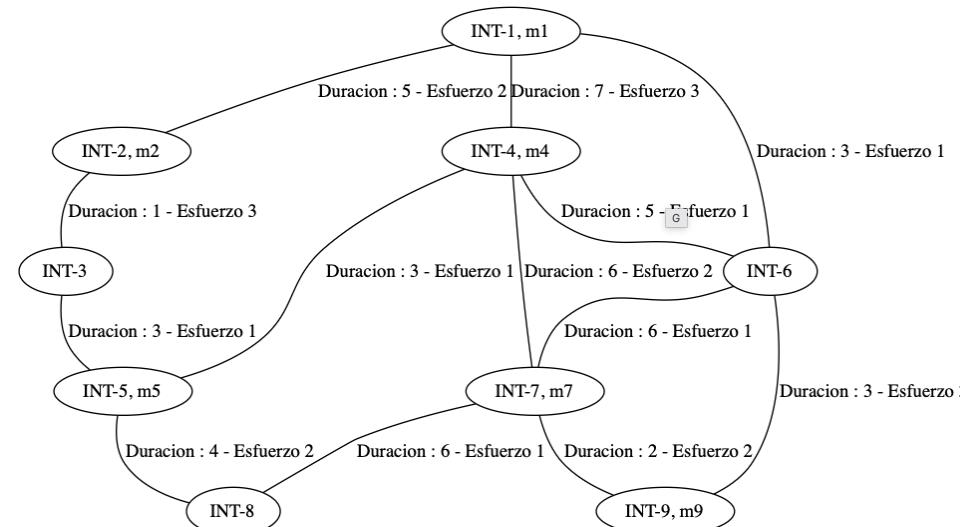
- Las restricciones del enunciado se traducen en la estructura de un nuevo grafo. → Construir **nuevo grafo** donde las aristas no representan colaboración, sino PROHIBICIÓN.
- Buscar el **número cromático** del nuevo grafo
- Cada color, una reunión

Ejercicio 4

Se tiene un grafo en el que las aristas son calles y los vértices son intersecciones entre calles. De cada arista se conoce la duración y el esfuerzo asociados con recorrer dicha calle a pie.

De cada vértice se conoce si alberga o no un monumento de interés, y en caso de que sí, cuál es el nombre y la relevancia de dicho monumento (valor entero en $[0,5]$).

Se está preparando una visita guiada por dichos monumentos para un grupo de alumnos de primaria, en la cual el desplazamiento entre los distintos monumentos se llevará a cabo a pie.

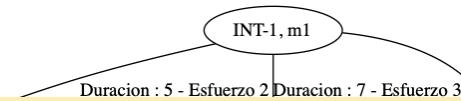


Ejercicio 4

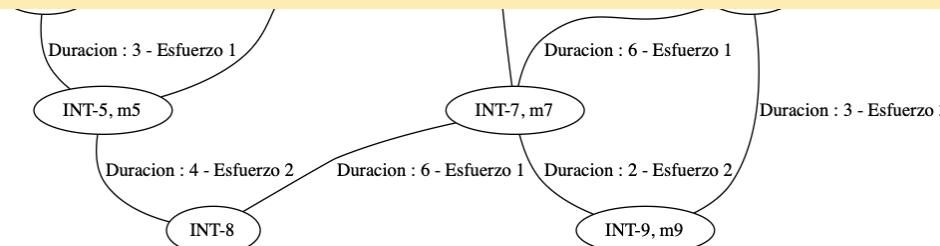
Se tiene un grafo en el que las aristas son calles y los vértices son intersecciones entre calles. De cada arista se conoce la duración y el esfuerzo asociados con recorrer dicha calle a pie.

De cada vértice se conoce si alberga o no un monumento de interés, y en caso de que sí, cuál es el nombre y la relevancia de dicho monumento (valor entero en [0,5]).

Se está preparando una visita guiada por dichos monumentos para un grupo de alumnos de primaria, en la cual el desplazamiento entre los distintos monumentos se llevará a cabo a pie.



Tengo dos atributos que pueden funcionar como peso, en un apartado necesitaré uno y en otro, otro.



Ejercicio 4a

- a) ¿Cuál es el **camino** de menor duración para ir de un monumento a otro dados?

Muestre el grafo configurando su apariencia de forma que se resalten los caminos.

toDot del grafo original

Ejercicio 4a

- a) ¿Cuál es el camino de menor duración para ir de un monumento a otro dados?

Muestre el grafo configurando su apariencia de forma que se resalten los caminos.

Pistas:

- Las aristas deben tener la duración como peso
- ¿Cuál es el algoritmo clásico para encontrar el camino más corto desde un único origen a todos los demás vértices en un grafo con pesos positivos? (Dijkstra)

Ejercicio 4b

b) Determine la ruta de menor esfuerzo que pase por todas las intersecciones exactamente una vez y vuelva al origen.

Muestre el grafo configurando su apariencia de forma que se resalte dicha ruta.

toDot del grafo original

Ejercicio 4b

b) Determine la ruta de menor esfuerzo que pase por todas las intersecciones exactamente una vez y vuelva al origen.

Muestre el grafo configurando su apariencia de forma que se resalte dicha ruta.

Pistas:

- Las aristas deben tener el esfuerzo como peso (`setEdgeWeight`)
- ¿Qué tipo de ciclo o ruta en la teoría de grafos cumple estas tres condiciones: 1) Visita todos los vértices, 2) Visita cada vértice una sola vez, y 3) Es un ciclo → Hamiltoniano

Ejercicio 4c

c) Dado un conjunto de calles cortadas al paso, determinar si se pueden seguir visitando *todos* los monumentos a pie *comenzando en uno cualquiera* de ellos. En caso contrario, determinar cuál es el conjunto de monumentos a visitar si se deben elegir los monumentos conectados entre sí que maximice la suma total de las relevancias.

Muestre el grafo configurando su apariencia de forma que se resalten dichos monumentos.



toDot del grafo original

Ejercicio 4c

c) Dado un conjunto de calles cortadas al paso, determinar si se pueden seguir visitando todos los monumentos a pie comenzando en uno cualquiera de ellos. En caso contrario, determinar cuál es el conjunto de monumentos a visitar si se deben elegir los monumentos conectados entre sí que maximice la suma total de las relevancias.

Muestre el grafo configurando su apariencia de forma que se resalten dichos monumentos.

Pistas:

- Las calles cortadas ya no son transitables. El análisis debe realizarse sobre las vías abiertas → (vista).
- Dividir un grafo en subconjuntos de vértices donde todos los nodos dentro del subconjunto son accesibles entre sí, pero no accesibles a nodos fuera de él → **COMPONENTES CONEXAS**
 - Si hay una única componente → FIN
 - Si hay más de una → elegir aquella que, al sumar la relevancia de sus monumentos, produzca el máximo valor total

**¡AHORA SÍ, ESTAMOS EN NUESTRA
HIGHWAY TO HELL!**



vevo