

PARTE 2: Ejercicios de aplicación

Duración: 60 minutos

Puedes tener **únicamente** material propio

EJERCICIO 1

Escenario

Los grafos **dirigidos** son un modelo matemático de gran aplicación en múltiples problemas de ruteo. Uno de los problemas típicos es, dada una red de conexiones dirigidas, y establecido un cierto nodo de esta red, determinar si desde el mismo es posible alcanzar a todos los demás o no.

Supongamos una gran red de vuelos aéreos de una cierta compañía de aviación. La aerolínea tiene vuelos entre diferentes ciudades o aeropuertos. Cada vuelo tiene atributos asociados, como ser identificador del vuelo, costo, tiempo de vuelo, etc. Los aeropuertos, a su vez, también tienen una cantidad de atributos, entre ellos su nombre.

El Gerente de Operaciones necesita contar con una funcionalidad, en su software de gestión operativa, que le permita, dado un cierto aeropuerto, indicar si es posible desde allí alcanzar a todos los demás aeropuertos de la red cubierta por la compañía. La funcionalidad debe ser lo más eficiente posible.

Consigna

Dado un **GRAFO DIRIGIDO**, representado por listas de adyacencias, desarrolla los siguientes algoritmos

En la clase **TGrafoDirigido**

De tipo boolean **alcanceTotal** (de tipo comparable nombreAeropuerto)

En la clase TVertice: firma libre

EJERCICIO 2

Escenario

Los grafos **NO DIRIGIDOS** son un modelo matemático de gran aplicación en múltiples problemas de conexión.

Una compañía de telecomunicaciones tiene una gran **red de datos** instalada en una cierta región. La red cuenta con “**servidores**” y “**enlaces**”.

Para enviar un paquete de datos desde un servidor a otro de la red es posible utilizar diferentes rutas. Un problema típico a resolver es encontrar una ruta entre dos servidores cualesquiera, que tenga la menor cantidad de saltos posible, es decir la mínima cantidad de servidores intermedios.

Por ello, se le ha encargado al Departamento de Sistemas que desarrolle una aplicación para que la consulta sobre cuál es la ruta entre dos servidores con la menor cantidad de saltos posible, se resuelva de la forma más eficiente posible.

El Arquitecto de Sistemas de la empresa ha determinado que como modelo para representar esta red de datos puede usarse un **GRAFO NO DIRIGIDO** (donde los servidores se representan mediante los vértices, y los enlaces mediante las aristas), y te ha encargado entonces que desarrolles y pruebes un método que permita obtener la ruta con menos saltos que enlace un cierto servidor con otro indicado.

Consigna

Dado un **GRAFO NO DIRIGIDO**, representado por listas de adyacencias, desarrolla un algoritmo con la siguiente firma, que se ejecute en un **orden del tiempo de ejecución lineal** respecto de la **cantidad de servidores** existentes en la red:

En la clase **TGrafoNoDirigido**

De tipo Lista <servidor> **rutaMenosSaltos** (de tipos comparables servidor1, servidor2) // devuelve la ruta de servidores con la menor cantidad de saltos, o nulo si la ruta no existe.

En la clase **TVertice**: firma libre

Nota: cada vértice posee un atributo de tipo vértice llamado “**predecesor**”, que **puede** ser usado para cargar allí el predecesor (anterior) a él en la ruta que se busca.