

EJERCICIO 1: Puntaje 5 puntos

Implemente la clase **Parcial**, y el método:

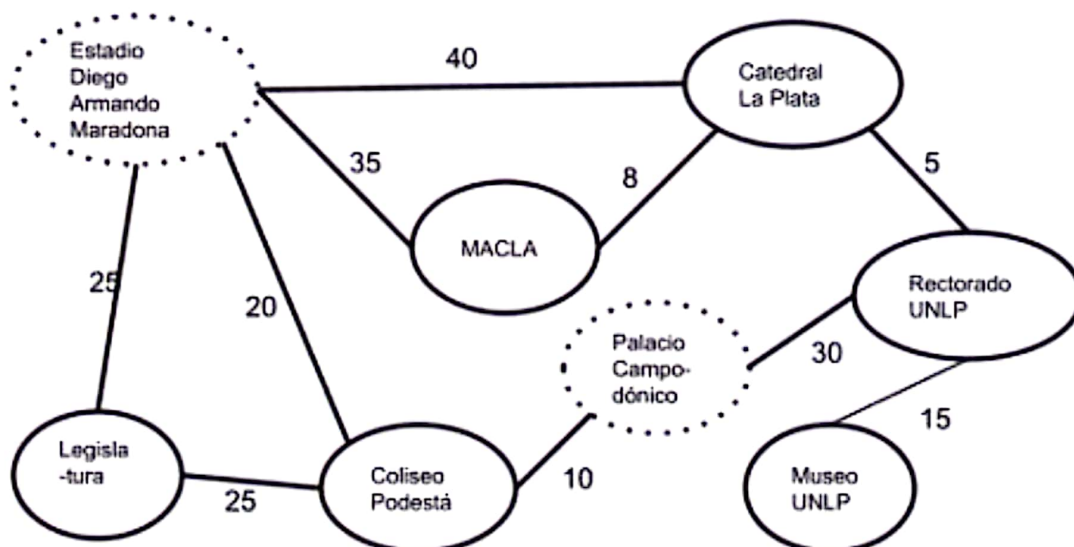
```
public ??? resolver(Grafo<???> sitios, String origen, String destino, ListaGenerica<???>
                    evitarPasarPor)
```

Los hinchas de las distintas selecciones que participan del mundial Sub-20 están visitando la ciudad de La Plata. Se quiere encontrar todos los caminos desde un sitio origen a un sitio destino evitando pasar por algunos sitios. Además, se quiere saber la distancia total recorrida para visitar todos los sitios de cada camino.

Cada vértice contiene el nombre del sitio y en las aristas se cuenta con la cantidad de cuadras que separan un sitio de otro.

Por ejemplo para este grafo, siendo el **origen** el **Estadio Unico "Diego Armando Maradona"**, el **destino** el **Palacio Campodónico** y los **sitios a evitar** son la **Legislatura** y **MACLA** la información a devolver sería:

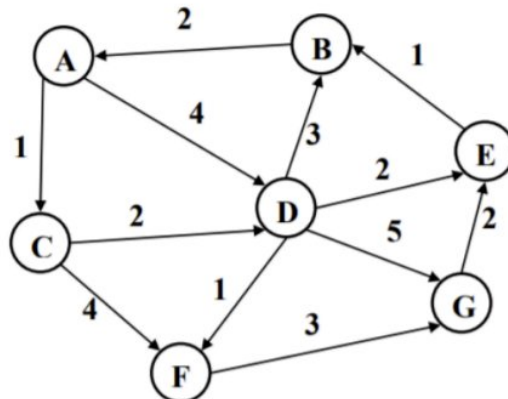
- **Estadio Unico "Diego Armando Maradona" - Coliseo Podestá - Palacio Campodónico. Distancia total = 30 cuadras.**
- **Estadio Unico "Diego Armando Maradona" - Catedral La Plata - Rectorado UNLP - Palacio Campodónico. Distancia total = 75 cuadras.**



- Completar en la firma del método los tipos de datos indicados con signo de interrogación.
- Debe verificar la existencia del origen y del destino.
- **No se puede pasar 2 veces por el mismo lugar** al formar el recorrido.
- En caso de no existir un recorrido posible, debe devolver la **lista vacía**.
- No se puede agregar variables de clase ni de instancia.

EJERCICIO 2: Puntaje 3 puntos

- a) Muestre paso a paso la ejecución de **Dijkstra** partiendo del vértice "**A**". Muestre todos los pasos intermedios, indicando el orden en que se van procesando los vértices.



Orden que toma el Vértice	Vértices	Distancia (A,v)	Vértice Previo	Visitado
	A	∞		0
	B	∞		0
	C	∞		0
	D	∞		0
	E	∞		0
	F	∞		0
	G	∞		0

- b) Marque en el grafo las aristas que forman el árbol con los caminos de costo mínimo que surgen del algoritmo.

EJERCICIO 3: Puntaje 2 puntos

Indicar cuál es la **ordenación topológica** para el siguiente grafo dirigido acíclico, utilizando la estrategia que trabaja con un **recorrido DFS** comenzando por el vértice A. Muestre paso a paso la ejecución del algoritmo.

Nota: considere que las listas de adyacentes están **ordenadas alfabéticamente** en forma **ascendente**.

