



Trabajo Práctico Obligatorio - Programación II

Título del TPO: Elección de ruta más rápida para una ambulancia en la ciudad

Alumnos:

Lucas Manzano LU: 1183893

Santiago Valentino Rossi LU: 1188973

Lola Cuartas LU: 1190036

Victoria Czapos LU: 1194118

Simon Romano LU: 1186529

Profesor: Perez, Nicolás Ignacio

Carrera: Ingeniería en Informática

Universidad: UADE – Universidad Argentina de la Empresa

Fecha de Entrega: 27 / 06 / 2025

Introducción

En situaciones de emergencia, cada minuto cuenta. La rapidez con la que una ambulancia puede llegar desde su base hasta un hospital puede ser la diferencia entre salvar una vida o no. Por eso, es fundamental contar con herramientas computacionales que permitan elegir la ruta más rápida en base a factores fundamentales, considerando el estado de las calles, el tiempo estimado de recorrido y la distribución geográfica.

Este trabajo muestra la optimización de una ruta más rápida para una ambulancia en una ciudad, modelando el mapa urbano como un grafo dirigido y ponderado, y aplicando el algoritmo de Dijkstra para hallar el camino más corto desde la base de ambulancia hasta un hospital central. De esta forma, generamos una herramienta que reduce el riesgo de empeoramiento de la salud de personas con urgencias.

Descripción del problema real

En una ciudad pequeña, existe una única base de ambulancia y un único hospital central. Ante una emergencia, la ambulancia debe recorrer el camino más corto en tiempo hacia el hospital. El desconocimiento del estado de las calles y tomar un camino que no se conoce puede generar demoras fatales para la vida de una persona en peligro.

La ciudad se modela como un grafo donde:

- Cada nodo representa una intersección o punto de referencia.
- Cada arista representa una calle entre dos intersecciones, con un peso que indica el tiempo estimado (en minutos) que toma recorrerla.

El objetivo es determinar automáticamente la ruta más rápida desde la base hasta el hospital, ayudando a reducir tiempos de traslado, evitar demoras innecesarias y optimizar la logística de emergencias médicas.

Algoritmo seleccionado: Dijkstra

El algoritmo de Dijkstra es ideal para este tipo de problemas donde se necesita encontrar la ruta más corta desde un nodo origen hacia otro destino dentro de un grafo con pesos no negativos.

Al aplicarlo nuestro programa funciona de la siguiente forma:

1. Inicializa la distancia a todos los nodos como infinita, excepto el nodo origen, que tiene distancia 0.
2. Usa una **cola de prioridad** para elegir siempre el nodo con la menor distancia no procesada.
3. Actualiza las distancias a sus vecinos si encuentra un camino más corto.
4. Repite hasta procesar todos los nodos alcanzables.

Ventajas para este problema:

- Permite hallar rutas en tiempo real y con alta eficiencia.
- Su uso con tiempos en minutos como pesos permite modelar el problema de forma precisa.
- Este algoritmo es determinista y confiable.

Implementación bajo TDA

Para representar el grafo y los elementos de la ciudad, se diseñaron los siguientes módulos e interfaces:

- **Nodo:** Representa una intersección, con su identificador, vecinos conectados y pesos asociados.
- **Grafo:** Contiene todos los nodos y métodos para agregar nodos y calles (aristas).
- **Interfaces (INodo, IGrafo):** Establecen la estructura base del sistema, facilitando la modularidad y extensibilidad.
- **Servicio Dijkstra:** Implementa el algoritmo de forma eficiente, utilizando una cola de prioridad (priority queue) , y almacenando la ruta completa y el tiempo total.

Este diseño modular y desacoplado permite una mejor organización del código, fácil mantenimiento, y claridad lógica entre los componentes.

Resultados y testeo

Implementamos una prueba en la clase TestGrafos con la siguiente configuración:

- **Nodos:** 0 a 5
- **Nodo 0:** Base de la ambulancia
- **Nodo 5:** Hospital central
- **Pesos de las aristas:** tiempo estimado en minutos

Tabla de calles simuladas:

Origen	Destino	Tiempo (min)
0	1	10
0	2	3
1	2	1
1	3	2
2	3	8
2	4	2
3	5	7
4	3	5
4	5	1

Resultado:

El sistema calcula correctamente la ruta más rápida y su tiempo total desde la base (nodo 0) al hospital (nodo 5), mostrando:

- **Ruta más rápida:** [0, 2, 4, 5]
- **Tiempo total estimado:** 6 minutos

Esto nos demuestra que el algoritmo es capaz de elegir rutas no obvias (por ejemplo, evitando calles con mayor tiempo aunque el recorrido sea más corto en distancia).

Comparación con otros algoritmos

- **A***: Más eficiente con heurísticas, pero requiere información adicional como distancia en línea recta, lo que no siempre está disponible. Ideal para mapas reales con GPS.
- **Floyd-Warshall**: Calcula todas las rutas entre todos los pares de nodos, pero no es necesario en este problema donde solo importa una ruta entre dos puntos.
- **Kruskal / Prim**: Diseñados para árboles de expansión mínima, sirven para construir redes óptimas pero no para encontrar rutas específicas.
- En este caso, Dijkstra sigue manteniéndose como mejor opción.

Conclusión

El uso del algoritmo de Dijkstra nos permitió resolver con eficacia un problema real en la logística médica urbana. Su implementación dinámica basada en TDA brinda claridad, reutilización y mantenimiento del código. Generamos una herramienta útil que evita demoras, agiliza procesos y cuenta con una interfaz amigable para el que la utilice.

La simulación mostró cómo la ambulancia puede encontrar la mejor ruta posible hacia el hospital, lo que en la vida real puede salvar vidas al reducir los tiempos de movilidad.