

Problema de Caminos Mínimos – Grupo 1

Algoritmo de Dijkstra

Problema: Una empresa de transporte necesita encontrar la ruta más corta entre su depósito y varias tiendas.

El objetivo principal es:

- **Minimizar el coste de transporte entre puntos (distancia o tiempo).**
- **Obtener caminos optimizados para planificar rutas eficientemente.**

Esto se puede modelar mediante un grafo ponderado, donde los nodos representan ubicaciones y las aristas representan rutas con un peso asociado (distancia o tiempo).

Representación TDA

Para implementar este problema se utilizaron los siguientes TDA:

TDA	Descripción
Nodo<T>	Representa un punto (depósito, tienda o pueblo).
Arista<T>	Representa un arco entre dos nodos con un peso asociado.
GrafoPesado<T>	Contiene la lista de nodos y la matriz de adyacencia con los pesos de las aristas.
Dijkstra	Algoritmo que calcula los caminos mínimos desde un nodo origen a todos los demás.

Algoritmo Dijkstra

Dijkstra permite calcular los caminos de menor costo desde un nodo origen en toda la red de nodos. El procedimiento básico es:

- 1. Inicializar todos los nodos con distancia infinita.**
- 2. Seleccionar el nodo origen y asignarle distancia 0.**
- 3. Considerar el nodo seleccionado como actual.**
- 4. Marcar el nodo como visitado y actualizar las distancias de sus vecinos.**
- 5. Repetir el proceso hasta visitar todos los nodos.**

El resultado final es un arreglo de distancias mínimas desde el nodo origen a cada nodo del grafo.

Comparación con otros algoritmos de grafos

Algoritmo	Propósito
Dijkstra	Calcula el camino mínimo desde un nodo origen a todos los demás en grafos ponderados sin aristas negativas.
Prim	Construye un árbol de expansión mínima conectando todos los nodos con el menor costo posible.
Kruskal	Selecciona los tramos con menor costo hasta unir todos los nodos, evitando ciclos, para formar un árbol mínimo.

Diferencia clave: Dijkstra encuentra caminos mínimos desde un origen específico, mientras que Prim y Kruskal construyen árboles de costo mínimo para todo el grafo.

Ejemplo de ejecución

Se construyó un grafo ponderado con 4 nodos:

<i>Nodo</i>	<i>Índice</i>
<i>Depósito</i>	<i>0</i>
<i>Tienda A</i>	<i>1</i>
<i>Tienda B</i>	<i>2</i>
<i>Tienda C</i>	<i>3</i>

Aristas con peso:

<i>Desde</i>	<i>Hacia</i>	<i>Peso</i>
<i>0</i>	<i>1</i>	<i>10</i>
<i>0</i>	<i>2</i>	<i>15</i>
<i>1</i>	<i>3</i>	<i>12</i>
<i>2</i>	<i>3</i>	<i>10</i>

Resultado del algoritmo Dijkstra desde el Depósito (nodo 0):

<i>Nodo</i>	<i>Distancia mínima desde Depósito</i>
<i>Depósito</i>	<i>0</i>
<i>Tienda A</i>	<i>10</i>
<i>Tienda B</i>	<i>15</i>
<i>Tienda C</i>	<i>22</i>

Esto confirma que el algoritmo calcula correctamente los caminos mínimos en un grafo ponderado.