

Centro de Enseñanza Técnica Industrial  
Inteligencia Artificial



6ºE1

Victor Abel Moreno Magaña 21110345

## ¿Qué es el algoritmo de Dijkstra?

El algoritmo de Dijkstra es un algoritmo utilizado en teoría de grafos y en ciencias de la computación para encontrar el camino más corto entre un nodo de inicio (o fuente) y todos los demás nodos en un grafo ponderado con pesos no negativos. Fue desarrollado por el científico de la computación holandés Edsger W. Dijkstra en 1956.

## ¿Para qué sirve el algoritmo de Dijkstra?

El algoritmo de Dijkstra se utiliza en una amplia gama de aplicaciones, que incluyen:

- **Redes de transporte y logística:** El algoritmo de Dijkstra se utiliza para calcular las rutas más cortas en redes de carreteras, ferrocarriles o vuelos. Las empresas de transporte y logística pueden utilizarlo para optimizar rutas de entrega o planificar itinerarios eficientes.
- **Redes de comunicación:** En el diseño y gestión de redes de comunicación, como Internet o redes telefónicas, el algoritmo de Dijkstra puede ayudar a encontrar las rutas más eficientes para transmitir datos o comunicaciones.
- **Sistemas de navegación:** Las aplicaciones de navegación, como los sistemas GPS, utilizan el algoritmo de Dijkstra para calcular las rutas más cortas y rápidas entre ubicaciones.
- **Diseño de circuitos integrados:** En la electrónica digital, el algoritmo de Dijkstra se utiliza en el diseño de circuitos integrados para optimizar las conexiones entre componentes en un chip, minimizando las distancias y retrasos.
- **Optimización en operaciones:** En la gestión de operaciones y la planificación de proyectos, el algoritmo de Dijkstra puede ser utilizado para encontrar la secuencia más corta de tareas o actividades.
- **Redes sociales y análisis de grafos:** En el análisis de redes sociales y otros tipos de análisis de grafos, el algoritmo de Dijkstra puede ser utilizado para encontrar la distancia o influencia entre nodos en una red.
- **Problemas de enrutamiento en general:** Cualquier problema que implique encontrar el camino más corto o la distancia mínima en un grafo ponderado con pesos no negativos puede beneficiarse del uso del algoritmo de Dijkstra.

## ¿Cómo se implementa el algoritmo de Dijkstra en el mundo?

El algoritmo de Dijkstra se implementa en una amplia gama de productos y servicios, que incluyen:

- **Sistemas de Navegación GPS:** Las aplicaciones de navegación, como Google Maps, Waze y sistemas GPS de automóviles, implementan el algoritmo de Dijkstra para encontrar rutas óptimas entre ubicaciones. Los conductores utilizan estas aplicaciones para recibir indicaciones precisas y rutas más cortas.
- **Redes de Transporte Público:** Las redes de transporte público, como sistemas de metro, autobuses y trenes, utilizan el algoritmo de Dijkstra para calcular las rutas más eficientes para los pasajeros, minimizando el tiempo de viaje y las transferencias.
- **Ruteo en Redes de Computadoras:** En el ámbito de las redes informáticas, los routers y switches pueden utilizar versiones adaptadas del algoritmo de Dijkstra, como el algoritmo OSPF (Open Shortest Path First), para determinar las rutas más cortas para el flujo de datos a través de una red.
- **Diseño de Circuitos Electrónicos:** En la industria de la electrónica, se puede utilizar el algoritmo de Dijkstra en el diseño de circuitos integrados para optimizar las conexiones entre componentes y minimizar las distancias entre ellos.
- **Planificación de Rutas en Entornos Logísticos:** Empresas de logística y transporte utilizan el algoritmo de Dijkstra para planificar rutas de entrega eficientes, lo que puede reducir costos y tiempos de entrega.
- **Optimización de Operaciones:** En operaciones y gestión de proyectos, el algoritmo de Dijkstra se utiliza para planificar tareas y actividades de manera eficiente, minimizando costos y tiempos de ejecución.
- **Redes Sociales y Análisis de Datos:** En el análisis de redes sociales y en el procesamiento de grandes conjuntos de datos, el algoritmo de Dijkstra se utiliza para calcular la distancia o la influencia entre nodos en una red.

## ¿Cómo implementaría el algoritmo de Dijkstra en mi vida?

Podría implementar el algoritmo para encontrar la ruta de camiones más eficiente de un lugar a otro y cuánto costaría.

## ¿Cómo implementaría el algoritmo de Dijkstra en mi trabajo o en mi trabajo de ensueño?

Podría implementar el algoritmo para encontrar el mejor camino para la entrega de baquetas

En mi trabajo de ensueño, podría utilizar el algoritmo de Dijkstra para ayudar a las personas a encontrar el camino más corto para acceder a servicios esenciales, como atención médica o educación.

## Código

```
import heapq

def dijkstra(graph, start):
    distances = {node: float('inf') for node in graph}
    distances[start] = 0
    previous_nodes = {}
    priority_queue = [(0, start)]

    while priority_queue:
        current_distance, current_node = heapq.heappop(priority_queue)

        if current_distance > distances[current_node]:
            continue

        for neighbor, weight in graph[current_node].items():
            distance = current_distance + weight

            if distance < distances[neighbor]:
                distances[neighbor] = distance
                previous_nodes[neighbor] = current_node
                heapq.heappush(priority_queue, (distance, neighbor))

    return distances, previous_nodes

def shortest_path(graph, start, end):
    distances, previous_nodes = dijkstra(graph, start)
    path = []
    while end:
        path.insert(0, end)
        end = previous_nodes.get(end)
    return path
```

```
graph = {
    'A': {'B': 1, 'C': 4},
    'B': {'A': 1, 'C': 2, 'D': 5},
    'C': {'A': 4, 'B': 2, 'D': 1},
    'D': {'B': 5, 'C': 1}
}

start_node = 'A'
end_node = 'D'
distances, previous_nodes = dijkstra(graph, start_node)

current_node = end_node
while current_node:
    print(f'Nodo actual: {current_node}, Distancia desde {start_node}: {distances[current_node]}')
    current_node = previous_nodes.get(current_node)

print(f'Camino mas corto desde {start_node} a {end_node}: {shortest_path(graph, start_node, end_node)}')
```

```
Nodo actual: D, Distancia desde A: 4
Nodo actual: C, Distancia desde A: 3
Nodo actual: B, Distancia desde A: 1
Nodo actual: A, Distancia desde A: 0
Camino mas corto desde A a D: ['A', 'B', 'C', 'D']
Press any key to continue . . . █
```

[https://github.com/VictorM8464/-6-E1\\_21110345\\_PR3.git?authuser=1](https://github.com/VictorM8464/-6-E1_21110345_PR3.git?authuser=1)

## **Bibliografía**

Navone, E. C. (2023). Algoritmo de la ruta más corta de Dijkstra - Introducción gráfica y detallada. freeCodeCamp.org.

<https://www.freecodecamp.org/espanol/news/algoritmo-> IA Zapopan, Jal.

México domingo, 22 de octubre de 2023 3

de-la-ruta-mas-corta-de-dijkstra-introducciongrafica/#:~:text=El%20algoritmo%20de%20Dijkstra%20encuentra,los%20dem%C3%A1s%20nodos%20del%20grafo