



**TECNOLÓGICO  
NACIONAL DE MÉXICO**



**Baez Saucedo Jesus Arnoldo**

**Prof. Jose Mario Rios Felix**

**Inteligencia Artificial**

**Tarea 3**

**18 de Septiembre de 2025**

## Algoritmo Dijkstra.

El algoritmo de Dijkstra es un método voraz que encuentra el camino más corto desde un nodo origen a todos los demás nodos en un grafo con pesos no negativos. Desarrollado por Edsger Dijkstra en 1956, utiliza una cola de prioridad para procesar nodos en orden de distancia creciente, relajando las aristas adyacentes y actualizando las distancias cuando encuentra caminos más cortos. Su complejidad es  $O((V+E) \log V)$  con cola de prioridad, donde  $V$  son vértices y  $E$  aristas. Es ampliamente usado en sistemas de navegación GPS, redes de computadoras, routing de paquetes y planificación de rutas por su optimalidad y eficiencia con pesos positivos.

```
import heapq

def dijkstra(grafo, inicio):
    distancias = {nodo: float('inf') for nodo in grafo}
    distancias[inicio] = 0
    cola = [(0, inicio)]

    while cola:
        dist_actual, nodo_actual = heapq.heappop(cola)
        if dist_actual > distancias[nodo_actual]:
            continue
        for vecino, peso in grafo[nodo_actual].items():
            distancia = dist_actual + peso
            if distancia < distancias[vecino]:
                distancias[vecino] = distancia
                heapq.heappush(cola, (distancia, vecino))
    return distancias

# Ejemplo
grafo = {
    'A': {'B': 1, 'C': 4},
    'B': {'A': 1, 'C': 2, 'D': 5},
    'C': {'A': 4, 'B': 2, 'D': 1},
    'D': {'B': 5, 'C': 1}
}

print("Distancias desde A:", dijkstra(grafo, 'A'))
```