



Algoritmo de Dijkstra

- **NOMBRE:** Luis Jorge Reyes González
- **REGISTRO:** 19310380
- **GRUPO:** 6E2
- **MATERIA:** Inteligencia Artificial
- **PROFESOR:** Mauricio Alejandro Cabrera Arellano

Algoritmo de Dijkstra

¿Qué es y cómo se usa?

El algoritmo de Dijkstra es un algoritmo utilizado para encontrar el camino más corto desde un nodo de origen a todos los demás nodos en un grafo ponderado y dirigido o no dirigido. Aquí está el funcionamiento básico del algoritmo de Dijkstra:

- 1. Inicialización:** Se asigna una distancia inicial de 0 al nodo de origen y una distancia infinita a todos los demás nodos. También se crea un conjunto de nodos no visitados.
- 2. Selección del nodo más cercano:** Se elige el nodo con la distancia más corta que aún no ha sido visitado y se marca como visitado.
- 3. Actualización de las distancias:** Para cada nodo adyacente al nodo seleccionado, se calcula la distancia acumulada sumando la distancia del nodo seleccionado más el peso de la arista que los conecta. Si esta distancia acumulada es menor que la distancia actualmente registrada para el nodo adyacente, se actualiza la distancia.
- 4. Repetición:** Se repiten los pasos 2 y 3 hasta que se hayan visitado todos los nodos o hasta que no queden nodos accesibles desde el nodo de origen.

Al finalizar, se obtiene la distancia más corta desde el nodo de origen hasta cada uno de los nodos en el grafo.

El algoritmo de Dijkstra utiliza una estructura de datos llamada "cola de prioridad" para seleccionar eficientemente el nodo con la distancia más corta en cada iteración. Esto asegura que siempre se seleccionen los nodos más cercanos primero.

El algoritmo de Dijkstra es óptimo para grafos con pesos no negativos. Sin embargo, si hay aristas con pesos negativos, se recomienda utilizar otros algoritmos como el algoritmo de Bellman-Ford.

Donde se usa

El algoritmo de Dijkstra se implementa en muchos campos y aplicaciones del mundo actual. Aquí hay algunos ejemplos de cómo se utiliza:

Redes de transporte: El algoritmo de Dijkstra se utiliza para encontrar las rutas más cortas en sistemas de navegación GPS, aplicaciones de transporte público y planificación de rutas en general. Permite determinar la ruta óptima para llegar de un lugar a otro minimizando la distancia o el tiempo de viaje.

Redes de comunicación: El algoritmo de Dijkstra se utiliza en enrutadores de redes de computadoras para encontrar la ruta más corta para enviar paquetes de datos de un nodo a otro en una red. Ayuda a optimizar el flujo de datos y minimizar la congestión en la red.

Optimización de recursos: El algoritmo de Dijkstra se utiliza en la optimización de recursos en campos como la logística, la gestión de la cadena de suministro y la planificación de proyectos. Ayuda a determinar la ruta más eficiente para el transporte de bienes, la asignación de recursos y la programación de tareas.

Redes sociales: En las redes sociales, el algoritmo de Dijkstra se utiliza para encontrar la ruta más corta entre dos usuarios o para calcular la distancia social entre ellos. También se utiliza en algoritmos de recomendación y en análisis de influencia en redes sociales.

Sistemas de navegación autónoma: En el campo de la robótica y los vehículos autónomos, el algoritmo de Dijkstra se utiliza para planificar rutas seguras y eficientes. Ayuda a los vehículos autónomos a evitar obstáculos y tomar decisiones sobre qué camino seguir.

Como lo implementaría en mi vida

Aunque no sea de manera directa, el algoritmo del funcionamiento me permitiría definir mi día a día, en el sentido que pueda planear las paradas o cosas que tengo por hacer, y definir cual es el camino más corto para finalizar mis tareas o pendientes.

Es decir, sería algo como:

Definir en que orden debo realizar mis actividades diarias, de manera que mi agenda quede de la forma

Levantarse – Ir al trabajo – Pasar por despensa – Llevar la ropa a la lavandería – Escuela – Casa.

Y que la secuencia anterior me suponga el menor tiempo en que yo puedo realizar esas acciones.

Como lo implementaría en el trabajo de mis sueños

Investigando un poco sobre este tema, he visto que para el trabajo de Ingeniero de Software Embebidos es útil saber y utilizar scripts que permitan realizar tareas repetitivas en menor tiempo.

De esta manera, la idea es buscar el tiempo que toma hacer cada prueba en las pruebas del hardware embebido.

Definir el tiempo que le toma partir de una prueba a otra.

Usar el algoritmo para generar los scripts en secuencia de tal manera que el tiempo total de la prueba sea el menor.