



TSU en Software

Práctica 3: implementación del algoritmo Dijkstra en Java.

Desarrollo de Aplicaciones II

Carlos Araiza Dionicio

Implementar el algoritmo de Dijkstra en Java de forma limpia y ordenada. Analizar la complejidad del algoritmo

Código fuente:

<https://github.com/iCharlieAraiza/DDAII/tree/main/Pr%C3%A1ctica%203>

Dijkstra.java

```
package com.coding;

public class Dijkstra {

    public int dijkstra(int[][] grafo, int nodoOrigen, int nodoFinal){
        int contNodo = grafo.length;
        boolean[] nodoVisitado = new boolean[contNodo];
        int[] distancia = new int[contNodo];

        for (int i = 0; i < contNodo; i++){
            nodoVisitado[i] = false;
            distancia[i] = Integer.MAX_VALUE;
        }
        distancia[nodoOrigen] = 0;
        for (int i = 0; i < contNodo; i++){
            int u = encontrarDistanciaMin(distancia, nodoVisitado);
            nodoVisitado[u] = true;
            for (int v = 0; v < contNodo; v++){
                if (!nodoVisitado[v] && grafo[u][v] != 0 && (distancia[u] + grafo[u][v] < distancia[v])){
                    distancia[v] = distancia[u] + grafo[u][v];
                }
            }
        }

        for (int i = 0; i < distancia.length; i++){
            if (i == nodoFinal)
                return distancia[i];
        }
        return 0;
    }
}
```

```

private int encontrarDistanciaMin(int[] distance, boolean[] visitedVertex) {
    int distanciaMin = Integer.MAX_VALUE;
    int distanciaMinNodo = -1;
    for (int i = 0; i < distance.length; i++){
        if(!visitedVertex[i] && distance[i] < distanciaMin){
            distanciaMin = distance[i];
            distanciaMinNodo = i;
        }
    }
    return distanciaMinNodo;
}
}

```

Analizar complejidad algorítmica.

Generalmente este tipo de algoritmos son del tipo $O(n \log n)$ ó $O(n \log m)$, ya que se hacen Pop y Add en el priority queue dentro de un ciclo de tipo $O(n)$, es decir, $O(\log n)$.

En el caso de Dijkstra, la complejidad en tiempo de ejecución es $O(|V| + |E| \log |e|)$.

Referencias

- Dijkstra's Algorithm: Explanation, Examples & Code. (2017, 21 mayo). [Vídeo]. YouTube. <https://www.youtube.com/watch?v=FSm1zybdOTk>
- D. (2018). Dijkstra Algorithm Implementation in Java | DevGlan. devglan. <https://www.devglan.com/datastructure/dijkstra-algorithm-java>