

Reporte: algoritmo Dijkstra

Para esta unidad solo disponemos de un solo código: “Dijkstra”; a lo largo del curso hemos planteado principalmente el denominado “problema del viajero” ¿en que consiste? Se plantea a un viajero entregar ciertos paquetes a diferentes destinos presentando varias rutas posibles, el problema radica en saber cual de todas estas mismas rutas posibles sería mas corta, es aquí donde entra el código dijkstra que de una manera un tanto compleja te arroja directa y fácilmente la ruta mas sencilla, corta, o menos costosa según sea el caso, o la solución que busquemos.

Pseudocódigo:

```
DIJKSTRA (Grafo  $G$ , nodo_fuente  $s$ )
    para  $u \in V[G]$  hacer
        distancia[ $u$ ] = INFINITO
        padre[ $u$ ] = NULL
        visto[ $u$ ] = false
    distancia[ $s$ ] = 0
    adicionar (cola, ( $s$ , distancia[ $s$ ]))
    mientras que cola no es vacía hacer
         $u$  = extraer_mínimo(cola)
        visto[ $u$ ] = true
        para todos  $v \in \text{adyacencia}[u]$  hacer
            si no visto[ $v$ ] y distancia[ $v$ ] > distancia[ $u$ ] + peso ( $u$ ,
v) hacer
                distancia[ $v$ ] = distancia[ $u$ ] + peso ( $u$ ,  $v$ )
                padre[ $v$ ] =  $u$ 
                adicionar(cola, ( $v$ , distancia[ $v$ ]))
```

Ejecución del código:

```
g=grafo()
g.conecta('a','b', 2)
g.conecta('b','c', 1)
g.conecta('c','d', 1)
g.conecta('d','e', 2)
g.conecta('e','f', 1)
g.conecta('a','i', 1)
```

```

g.conecta('i','c', 4)
g.conecta('d','g', 1)
g.conecta('g','e', 2)
g.conecta('g','h', 1)
g.conecta('h','f', 2)
g.conecta('i','j', 1)
g.conecta('j','k', 1)
g.conecta('k','u', 2)
g.conecta('k','l', 1)
g.conecta('u','t', 1)
g.conecta('l','m', 1)
g.conecta('m','d', 3)
g.conecta('m','n', 1)
g.conecta('n','g', 3)
g.conecta('n','o', 1)
g.conecta('o','h', 3)
g.conecta('l','p', 1)
g.conecta('p','t', 1)
g.conecta('p','q', 1)
g.conecta('q','n', 2)
g.conecta('q','s', 1)
g.conecta('t','s', 1)
g.conecta('s','r', 1)
g.conecta('r','o', 3)
print(g.shortest('f'))

```

Antes de hacer la llamada a quien vendría ser el código dijkstra en este caso denominado como “shortest” se deben de crear los grafos, que vendrían a jugar el papel de dichas puntos de encuentro a quienes debemos visitar dentro de las rutas posibles, para este ejemplo que cree me basé en la ruta que podría tomar desde el punto “a” que es mi casa al punto “f” que sería la parada del camión, una vez que conectamos todos los nodos del grafo mediante aristas, podemos correr el algoritmo de dijkstra que arrojaría el camino mas corto a seguir que de manera gráfica se observaría que es la ruta: a-b-c-d-e-f básicamente.