```
import math

INFINI = math.inf
NEIGHBOURS = "neighbours"

def dijkstra(graph, origin, destination=None):
    distances = {}
    ancestors = {}
    unknown = {}
    path = []
    finish = False
    for city in graph:
        unknown[city] = True
        ancestors[city] = origin
        if city in graph[origin][NEIGHBOURS]:
            distances[city] = graph[origin][NEIGHBOURS][city]
        else:
            distances[city] = INFINI
    distances[origin] = 0
    unknown[origin] = False
    new_city = origin
    while not finish:
        minimum = INFINI
        for city in graph:
            if unknown[city] and distances[city] < minimum:
                new_city = city
                minimum = distances[city]
        if minimum == INFINI:
            finish = True
        unknown[new_city] = False
        for town in graph[new_city][NEIGHBOURS]:
            if distances[new_city] + graph[new_city][NEIGHBOURS][town] <
distances[town]:
                distances[town] = distances[new_city] +
graph[new_city][NEIGHBOURS][town]
                ancestors[town] = new_city
    if (destination is not None) and (distances[destination] != INFINI):
        city = destination
        while city != origin:
            path.append(city)
            city = ancestors[city]
        path.append(origin)
        path.reverse()
    return [distances, path]
```

From:
https://wiki.centrale-med.fr/informatique/ - **WiKi informatique**

Permanent link:
https://wiki.centrale-med.fr/informatique/jour_tp_chemin:dijjstra

Last update: **2018/11/16 14:43**