

## Algorytmy i Struktury Danych II, Zestaw 7

### ALGORYTM DIJKSTRY

Wykorzystując algorytm Dijkstry [Link 1, Link 2], proszę napisać program znajdujący najkrótszą drogę pomiędzy dwoma węzłami w grafie skierowanym z wagami nieujemnymi. Działanie algorytmu Dijkstry tłumaczy poniższy pseudokod:

```

DIJKSTRA( $G, w, s$ )
1  INITIALIZE-SINGLE-SOURCE( $G, s$ )
2   $S = \emptyset$ 
3   $Q = G.V$ 
4  while  $Q \neq \emptyset$ 
5       $u = \text{EXTRACT-MIN}(Q)$ 
6       $S = S \cup \{u\}$ 
7      for each vertex  $v \in G.Adj[u]$ 
8          RELAX( $u, v, w$ )
    
```

gdzie **EXTRACT-MIN** jest kolejką priorytetową typu min względem atrybutu  $d$ , natomiast

**INITIALIZE-SINGLE-SOURCE**( $G, s$ )

**FOR** każdy wierzchołek  $v \in G.V$ ,  $v.d = \infty$ ,  $v.\pi = \text{NIL}$ ,  $s.d = 0$

Relaksacja krawędzi jest realizowana za pomocą procedury

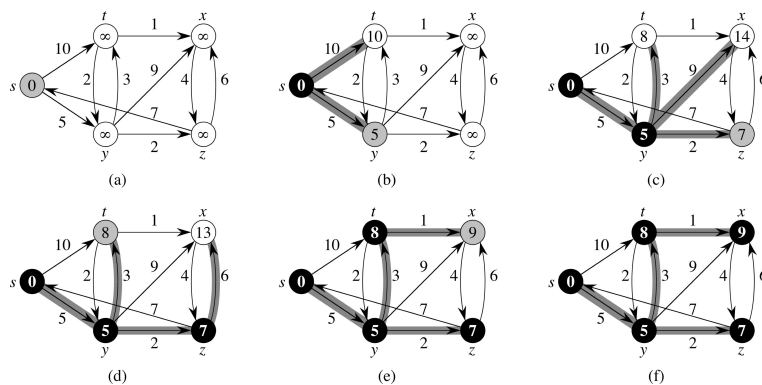
**RELAX**( $u, v, w$ )

**if**  $v.d > u.d + w(u, v)$

$v.d = u.d + w(u, v)$

$v.\pi = u$

gdzie  $v.\pi$  to poprzednik węzła  $v$ . Poniżej zaprezentowano przykład ilustrujący działanie algorytmu Dijkstry:



Rozpatrywany graf należy wczytać z dwóch plików tekstowych: 1) zawierającego macierz sąsiedztwa, 2) zawierającego wagi krawędzi.