Pseudocodice

K. Cammini minimi

K.1. Calcolo della lunghezza dei cammini mimimi tra ogni coppia di vertici

Algoritmo *FloydWarshall (grafo pesato* $G = (V, E, \omega)$*)* $\rightarrow$ *matrice distanze*

```
/* La funzione ω assegna un peso a ciascun arco.
   Il grafo non deve contenere cicli di peso negativo.          */
Sia D una matrice n × n, con n = #V
for i ← 1 to n do
    for j ← 1 to n do
        if i = j then D[i, j] ← 0
        else if (v_i, v_j) ∈ E then D[i, j] ← ω(v_i, v_j)
        else D[i, j] ← ∞
for k ← 1 to n do
    for i ← 1 to n do
        for j ← 1 to n do
            if D[i, k] + D[k, j] < D[i, j] then
                D[i, j] ← D[i, k] + D[k, j]
return D
```

K.2. Calcolo della lunghezza dei cammini mimimi da un vertice s a tutti gli altri

Algoritmo *Dijsktra (grafo pesato* $G = (V, E, \omega)$*, vertice* s*)* $\rightarrow$ *vettore distanze*

```
/* La funzione ω assegna un peso a ciascun arco.
   Il grafo non deve contenere archi di peso negativo.          */
Sia D un vettore con insieme di indici V
D[s] ← 0
foreach v ∈ V \ {s} do D[v] ← ∞
C ← V
while C ≠ ∅ do
    u ← elemento di C con D[u] minima
    C ← C \ {u}
    foreach (u, v) ∈ E do
        if D[u] + ω(u, v) < D[v] then
            D[v] ← D[u] + ω(u, v)
return D
```