

(**int**[], **int**[]) CamminiMinimi(**GRAPH** G , **NODE** s)

int[] d = new **int**[1... $G.n$]

// $d[u]$ è la distanza da s a u

int[] T = new **int**[1... $G.n$]

// $T[u]$ è il padre da u nell'albero T

// Inizializzo i vettori

foreach $u \in G.V - \{s\}$ **do**

$T[u] = \text{nil}$ $d[u] = +\infty$

// Inizializzo la sorgente

$T[s] = \text{nil}$ $d[s] = 0$

// Effettuo l'ordinamento topologico dei nodi nel DAG

STACK S = topSort

// fintanto che la pila non è vuota

while not $S.\text{isEmpty}$ **do**

$u = S.\text{pop}$ // estraggo un nodo

foreach $v \in G.\text{adj}(v)$ **do** // per ogni nodo adiacente

if $d[u] + G.w(u,v) < d[v]$ **then** // se il peso è migliore di quello presente

 // aggiorno il peso

$T[v] = u$ $d[v] = d[u] + G.w(u,v)$

// restituisco il vettore dei padri e il vettore delle distanze

return (T, d)