

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

(1)    PRIORITYQUEUE  $S \leftarrow$  PriorityQueue //  $\mathcal{O}(n) \cdot 1$   
 $S.inserisci(s, 0)$

(2)    finché not  $S.isEmpty()$  fai //  $\mathcal{O}(n)$

    //  $\mathcal{O}(n)$  vettore ordinato /  $\mathcal{O}(\log n)$  heap binario

**int**  $u \leftarrow S.deleteMin$

$b[u] \leftarrow$  falso

    per ciascun  $v \in G.adj(u)$  fai

        se  $d[u] + G.w(u,v) < d[v]$  allora

            se not  $b[v]$  allora

                //  $\mathcal{O}(1) \cdot n$  vettore ordinato /  $\mathcal{O}(\log n) \cdot n$  heap binario

$b[v] \leftarrow$  vero

            altrimenti

                //  $\mathcal{O}(1) \cdot m$  vettore ordinato /  $\mathcal{O}(\log n) \cdot m$  heap binario

$T[v] \leftarrow u$

$d[v] \leftarrow d[u] + G.w(u,v)$

ritorna  $(T, d)$