

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

```
(1)  QUEUE  $S \leftarrow$  Queue  
     $S.enqueue(s)$  // metto in coda la sorgente  
finché not  $S.isEmpty$  fai //  $\mathcal{O}(n)$   
(2)  | int  $u \leftarrow S.dequeue$  //  $\mathcal{O}(1 \cdot n)$   
    |  $b[u] \leftarrow$  falso  
    | per ciascun  $v \in G.adj(u)$  fai  
    | | se  $d[u] + G.w(u, v) < d[v]$  allora  
    | | | se not  $b[v]$  allora  
(3) | | | | // lo metto in coda quando c'è un miglioramento  
    | | | |  $S.enqueue(v)$  //  $\mathcal{O}(m \cdot n)$   
    | | | |  $b[v] \leftarrow$  vero  
    | | |  $T[v] \leftarrow u$   
    | |  $d[v] \leftarrow d[u] + G.w(u, v)$   
    | ritorna  $(T, d)$ 
```