

Sia  $G$  un grafo orientato e pesato :

- $d(v, u)$  distanza di vertice  $v$  a vertice  $u$
- peso di un cammino di peso min tra tutti quelli da  $u$  a  $v$  :  
 $d(v, u) = \min \{ W(p) \mid p \text{ e' un cammino } v \rightarrow u \}$   
dove  $W(p)$  e' la somma dei pesi degli archi del cammino
- $d(u, v)$  ben definito solo se nessun cammino  $v \rightarrow u$  contiene un ciclo di peso negativo

## ALGORITMO :

input  $\rightarrow$   $G$  orientato e pesato  
 $s$  nodo sorgente

output  $\rightarrow \forall v \in V \quad v.d = \text{distanza di } v \text{ da } s$   
 $\hookrightarrow$  mantiene una STIMA della distanza

All' inizio :  $s.d = 0 \quad \forall v \in V, v \neq s, v.d = \infty$

$\rightarrow$  Si costruisce l'albero con radice  $s$ , in cui viene inserito un vertice per volta

$\rightarrow$  l'albero e' memorizzato come insieme degli archi  $(v.\pi, v)$

$\rightarrow$  quando inserisco  $u$  : aggiorno le stime delle distanze dei vertici adiacenti

### Dijkstra( $G, s$ )

```
 $Q \leftarrow V$   
for  $\forall v \in V$  do  $v.d \leftarrow \infty, v.\pi \leftarrow nil$   
 $s.d \leftarrow 0$   
 $s.\pi \leftarrow nil$   
while  $Q \neq \emptyset$  do  
   $u \leftarrow$  toglì nodo con  $d$  minimo da  $Q$   
  for  $\forall v \in adj[u]$  do  
    if  $v \in Q$  e  $u.d + W(u, v) < v.d$  then  
       $v.d \leftarrow u.d + W(u, v)$   
       $v.\pi \leftarrow u$ 
```

se un nodo non e' raggiungibile da  $s$ ,  
mantiene  $v.d = \infty$  e  $s.\pi = nil$

$\rightarrow$  prendo il nodo se il cammino e' < della stima

$$O(|E| \log |V|)$$

