

Pseudocodice

J. Minimo albero ricoprente

J.1. Algoritmo di Kruskal (schema ad alto livello)

```
Algoritmo Kruskal (grafo  $G = (V, E)$  con pesi sugli archi)  $\rightarrow$  albero  
/* Costruzione di un albero di ricoprimento di peso minimo */  
ordina l'insieme  $E$  in base ai pesi in modo non decrescente  
 $T \leftarrow$  albero vuoto  
foreach  $(x, y) \in E$  secondo l'ordine do  
| if  $x$  e  $y$  non sono connessi in  $T$  then  
| | aggiungi a  $T$  l'arco  $(x, y)$   
return  $T$ 
```

J.2. Algoritmo di Kruskal (mediante Union-Find)

```
Algoritmo Kruskal (grafo  $G = (V, E)$  con pesi sugli archi)  $\rightarrow$  albero  
/* Costruzione di un albero di ricoprimento di peso minimo */  
ordina l'insieme  $E$  in base ai pesi in modo non decrescente  
 $T \leftarrow$  albero vuoto  
foreach  $v \in V$  do makeSet( $v$ )  
foreach  $(x, y) \in E$  secondo l'ordine do  
|  $t_x \leftarrow$  find( $x$ )  
|  $t_y \leftarrow$  find( $y$ )  
| if  $t_x \neq t_y$  then  
| | union( $t_x, t_y$ )  
| | aggiungi a  $T$  l'arco  $(x, y)$   
return  $T$ 
```

J.3. Algoritmo di Prim (schema ad alto livello)

```
Algoritmo Prim (grafo  $G = (V, E)$  con pesi sugli archi)  $\rightarrow$  albero  
/* Costruzione di un albero di ricoprimento di peso minimo */  
 $T \leftarrow$  albero formato da un solo nodo  $s$   
while  $T$  ha meno di  $n$  nodi do  
| trova un arco  $(x, y)$  di peso minimo con un estremo in  $T$  e l'altro non in  $T$   
| aggiungi a  $T$  l'arco  $(x, y)$   
return  $T$ 
```
