

Kruskal's Algorithm

Input: A weighted undirected graph $G = (V, E)$ with weight function $f : E \rightarrow \mathcal{R}^+$

Output: A minimal spanning tree T for G .

Method:

```
 $T = \phi$  ;  
 $S = \phi$  ;  
Construct a priority queue  $Q$  of all edges in  $E$  ;  
for each vertex  $v \in V$  do add  $\{v\}$  to  $S$ .  
while  $\|S\| > 1$  do  
    choose edge  $(v, w)$  from  $Q$  of lowest cost ;  
    delete  $(v, w)$  from  $Q$  ;  
    if (  $v$  and  $w$  are in different sets  $W_1$  and  $W_2$  in  $S$  ) then  
        Replace  $W_1$  and  $W_2$  in  $S$  by  $W_1 \cup W_2$  ;  
        Add  $(v, w)$  to  $T$  ;  
    end if  
end while
```