

## P2

---

### Prompt:

Given an undirected weighted graph  $G = (V, E)$ , such that the cost  $c_e > 0$  for all edges  $e \in E$  and for any two edges  $e, f \in E$ ,  $c_e \neq c_f$ , give a polynomial time algorithm to find the minimum cost set of edges  $F \subseteq E$  such that a graph  $G' = G - F$  has no cycles.

### Algorithm:

```
// Let graph  $G = (V, E)$  with edge weights  $w$ 
MaxSpanTree( $G, w$ ):
    Let  $E'$  be an empty set

    // Note, that the following sort, Sort  $E$ , denotes:
    // Sorting by the inverse of each weight in ascending order.
    // Ex: If  $E = [2, 4, 1, 3, 0]$  then Sort  $E = [-4, -3, -2, -1, 0]$ 
    Sort  $E$ 

    // Make each vertex a singleton set
    For all vertices  $v \in V$ 
        Set( $v$ )
    EndFor

    // Get the edges  $E'$  of the maximum spanning tree of  $G$ 
    For all edges  $\{u, v\} \in E$  in ascending order of weight
        If Set( $u$ )  $\neq$  Set( $v$ ) then
            Add edges  $\{u, v\}$  to  $E'$ 
            Union( $u, v$ )
        EndIf
    EndFor
    Return  $E'$ 

// Get the minimum cost edge set  $F$  of  $G$ 
Let  $F = E - E'$ 
```

---

### Claim:

The algorithm terminates in polynomial time.

### Proof:

We know that Kruskal's algorithm terminates in  $O(m \log n)$ . The algorithm is based on Kruskal's algorithm and only changes the order of the edges by weight. Because no other changes are made, the algorithm's runtime is unchanged. Thus, the algorithm terminates in  $O(m \log n)$ .  $\square$

---

**Claim:**

For any cycle  $C$  in the graph  $G$ , if the weight  $w_e < w_f$  for edge  $e \in C$  and all other edges  $f \in C$ , then  $e$  is not in a maximum spanning tree  $T$  of  $G$ .

**Proof:**

Assume for a contradiction, that in any cycle  $C$  in graph  $G$  there is an edge  $e \in C$  with a weight  $w_e < w_f$  for all other edges  $f \in C$  and that  $e$  is in the maximum spanning tree  $T$  of  $G$ . Removing  $e$  from  $T$  results in a disconnected graph. Because  $e$  was in  $C$ , there is an edge  $f \in C$  that can reconnect the graph. Let  $T' = T - e \cup f$ . Because  $w_e < w_f$  the weight of  $T < T'$ . Thus,  $T$  is not the maximum spanning tree of  $G$  which is a contradiction.  $\square$

---

**Claim:**

For any cut set  $C$  in the graph  $G$ , if the weight  $w_e > w_f$  for edge  $e \in C$  and all other edges  $f \in C$ , then  $e$  is in all maximum spanning trees  $T$  of  $G$ .

**Proof:**

Assume for a contradiction, that for any cut set  $C$  in the graph  $G$  the weight  $w_e > w_f$  for edge  $e \in C$  and all other edges  $f \in C$  and that  $e$  is not in all maximum spanning trees of  $G$ . Let  $T$  be one such maximum spanning tree of  $G$  that does not contain  $e$ . Adding  $e$  to  $T$  will form a cycle with another edge  $f$  across a cut which was previously only spanned by  $f$ . Removing  $f$  from  $T$  results in a new spanning tree  $T' = T - f \cup e$ . Because the weight  $w_e > w_f$  the weight of  $T' > T$ . Thus,  $T$  is not a maximum spanning tree of  $G$  which is a contradiction.  $\square$

---

**Claim:**

For any  $x, y \in \mathbb{R}$  if  $x < y$  then  $-x > -y$ .

**Proof:**

Assume for a contradiction, that there exists some pair  $x, y \in \mathbb{R}$  such that  $x < y$  and  $-x \leq -y$ . Let  $x$  be less than  $y$  by some positive, nonzero amount  $k$  such that  $x + k = y$ . Then, by substitution  $-x \leq -(x + k)$ . Algebra shows this to be equivalent to  $0 \geq k$ . But, since  $k$  is a positive, nonzero number, this is a contradiction.  $\square$

---

**Claim:**

Given an undirected weighted graph  $G = (V, E)$ , such that the cost  $c_e > 0$  for all edges  $e \in E$  and for any two edges  $e, f \in E$ ,  $c_e \neq c_f$  the algorithm finds the minimum cost set of edges  $F \subseteq E$  such that a graph  $G' = G - F$  has no cycles.

**Proof:**

We have already seen proved that Kruskal's algorithm finds the minimum spanning tree and that the algorithm relies on the ordering in which edges are sorted. Let  $G = (V, E)$  be a graph with edge weights  $w$ . We have shown that for any pair of values  $x, y$  if  $x < y$  that  $-x > -y$  after inverting. We apply this property and invert the weights in  $w$  and run Kruskal's algorithm. By the cut property, we know that the minimum spanning tree  $T$  produced by Kruskal's algorithm includes the minimum weighted edges at each cut. Let  $T^*$  denote the tree produced by Kruskal's algorithm with inverted weights. Because we've inverted the weights, the edges in  $T^*$  are the minimum weighted edges after inversion and so must have been the maximum weighted edges before inversion. Therefore  $T^*$  is the maximum spanning tree of  $G$  and by the definition of tree  $T^*$  has no cycles. Now, we also know by the cycle property, that for any cycle  $C$  in  $G$ , the maximum weighted edge  $f \in C$  will not be in  $T$ . So, the edges  $F$  that are not included in  $T^*$  are the maximum weighted edges after inversion and so must have been the minimum weighted edges before inversion. We know then, that running Kruskal's algorithm on  $G$  with inverted weighted edges produces a maximum spanning tree  $T^*$  and a set of minimum weighted edges  $F$ . Thus, we know that Kruskal's algorithm on inverted weighted edges produces  $T^* = G - F$ .  $\square$