Show all work. Unjustified answer yields no credit. You can use any part of this problem, without showing it, to show another part of this problem.

1. (5 pts) Let \( G_4 = (V, E) \) be a simple graph on 4 vertices obtained as follows. To a cycle \( C_3 = (V(C_3), E(C_3)) \) (on 3 vertices) add a vertex \( v \) which is adjacent to all vertices in \( V(C_3) \). How many spanning trees \( G_4 \) has?

2. (5 pts) Let \( G = (V, E) \) be a connected simple graph. Assign to each edge in \( e \in E \) a weight \( c(e) \in \mathbb{R} \). Assume that each two distinct edges \( e, f \in E \) have different weights: \( c(e) \neq c(f) \). Is a minimum spanning tree unique? If yes, give a short argument; if no, give a simple counter example.

\[ 2 \times 2 = 4 \]

**Proof**

In every step of Kruskal algo the any minimal tree \( T \) contains the forest \( F_k \) generated by Kruskal algo (Thm 2.33). Claim -

\( F_k \) is unique. Proof by induction. \( F_1 \) choose cheapest edge - unique since all weights are different.

Assume true for \( 1 \leq h \leq k \). Let \( h = k + 1 \). \( F_k \) has \( k \) connected components. Choose the cheapest edge connecting these components unique. Claim \( T \) must contain \( F_k \) for 1 each \( h \).
Proof: \( k = 1 \). Suppose \( T \) (minimal tree) does not contain \( e \), the cheapest edge.

Then \( T = T + e = T' \) is also minimal for some \( f \in E(T) \) if \( c(T') = c(T) + c(e) - c(f) < c(T) \) as \( c(e) \leq c(f) \) for \( f \neq e \), so \( T \) can not be minimal. Contradiction. Assume that \( T \) contains \( F_k \). So \( T \) connects the connected components of \( F_k \). Let

\[ E(F_k) \cup E(F_{k+1}) = E(F_{k+1}). \]

Suppose that \( f_{k+1} \notin E(T) \).

So \( f_{k+1} \notin T \). \( T + f_{k+1} f_{k+1} = T' \), minimum tree.

\( f_{k+1} \) also connects the components of \( F_k \). Hence since \( c(f_{k+1}) \) was the smallest from all edges connecting the connected components

\[ c(f_{k+1}) < c(f_{k+1}) \] (contradiction to the minimality of \( T \). Hence \( T \) contains \( F_1, F_2, \ldots, F_{k-1} = F_{k-1} = T \).