Proposition:

Longest MST edge. Run empirical studies to analyze the length of the longest edge in the MST and the number of graph edges that are not longer than that one.

Proof:

From the proposition, let's consider a graph with Vertices “V” and Edges “E” connected with weights. Where N is the number of edges less than longest edge in mst.

Procedure:

The above can be calculated by using certain procedure. First, let create graph **G** and at the same time insert the edge weights in the BInary Symbol table **bst**. Using the Kruskal’s algorithm find the mst and insert the edges of mst in pq, so that it is to obtain the longest edge in mst(***max***). It is to find the rank from symbol table by using getRank(). The rank represent the number of edges below the longest edge from the bst.

The time complexity from the kruskal algorithm to create a mst is ***E*logE.*** From the reference mention below (of Property 20.10) that MST of a graph in time proportional to ***E + N*log V.***

Data Analysis:

Data is taken from the program written using Certain graphs from test cases with weights.

The program can be obtained from the github:
https://github.com/kvmuralikrishna1993/ADS2/blob/master/assignments/MST.zip

V = Vertices of the graph
E = Edges of the graph
N = number of edges lower than longest edge in mst

TIme complexity ∝ ElogE

Proposed time Complexity ∝ E+N*logV

Ratio (R) = ElogE/E+N*logV

V	E	N	ElogE	E+N*logV	R
8	16	12	64	52	1.23
250	1273	961	13130	8928.1	1.47
9	14	6	53.3	33	1.61
4	5	1	11.6	7	1.66
3	3	0	4.8	3	1.6

Conclusion: From the above analysis the ratio is constant and approximately equals to ≅ **1.6** and the deviation increases with the edges equal to longest edge in the mst.

References: https://flylib.com/books/en/3.56.1.51/1/ (Property 20.10).