

SSN College of Engineering
Department of Computer Science and Engineering
CS1403 — Design and Analysis of Algorithms
2019 – 2020
Session — 08
February 12, 2020

- This homework is due by 4pm on February 12, 2020
 - Grace period may be given up to midnight of February 13, 2020
 - You can upload only one ZIP file
 - The naming convention is “<Your first name (first letter capital and all the other letters small)>-CS1403-S08.zip”
 - The questions marked as “OPTIONAL” are, as the name implies, optional! Complete your core assignment first and attempt the optional problem only if you have sufficient time.
-

1. Given a graph $G = (V, E)$, a **spanning tree** is a tree formed with all the vertices in V and a subset of edges $E_s \subseteq E$ s.t. all the vertices are covered. **Cost of a spanning tree** is the sum of the costs (weights) of the edges forming the tree.

Problem: Given a weighted graph $G = (V, E)$, find a minimum-cost spanning tree (MST) of G .

- (a) Implement an efficient data structure for maintaining equivalence classes (disjoint sets) of a set S based on certain equivalence relation. Provide implementations of *union* and *find* operations. Your *union* should run in constant time and *find* in $O(\log |S|)$ time. Implement **path compression** to make *find* run in near constant time.
- (b) Implement Kruskal’s algorithm with time complexity not exceeding $O(|E| \log |V|)$. You should use your union-find implementation to achieve this time complexity.