

Ryan Christopher

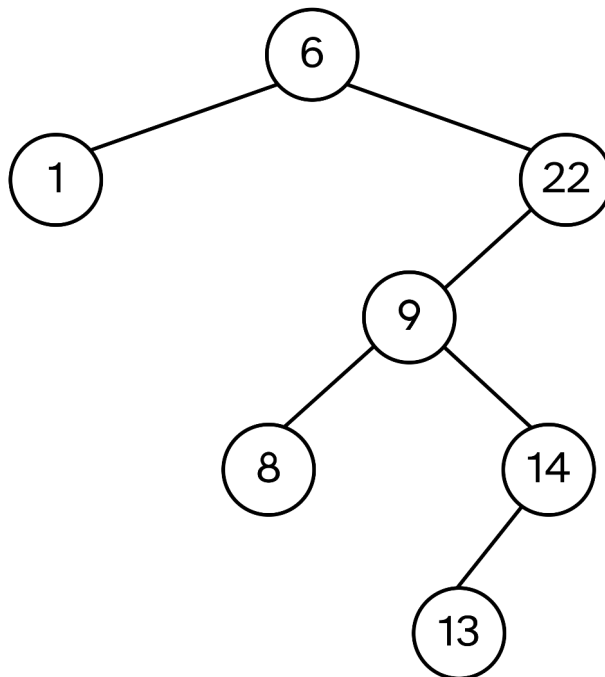
MET CS566

Assignment 4

1. Given the following key sequence:

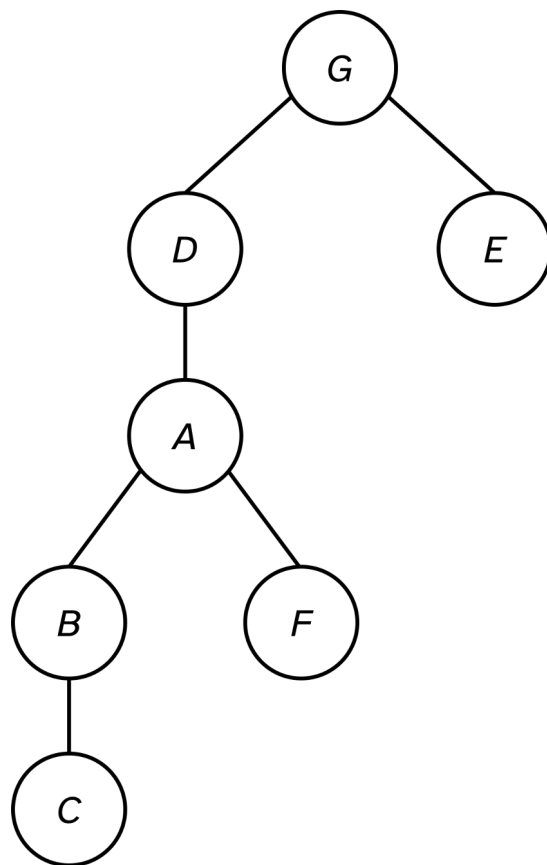
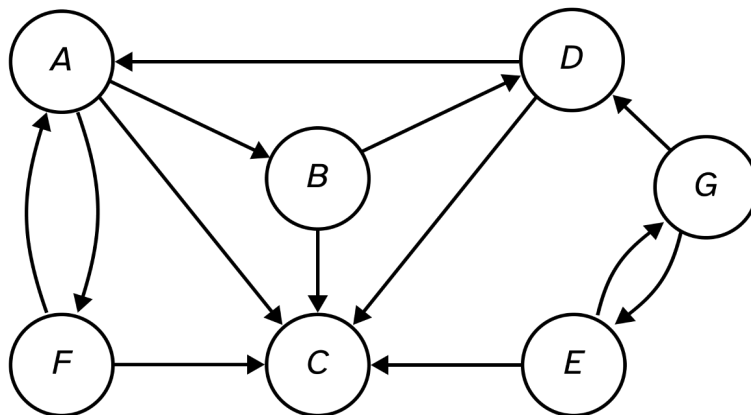
(6, 22, 9, 14, 13, 1, 8)

build the dynamic binary search tree **without balancing it**. How many probes (i.e. comparisons) does it take to determine that the key 100 is not in the tree?

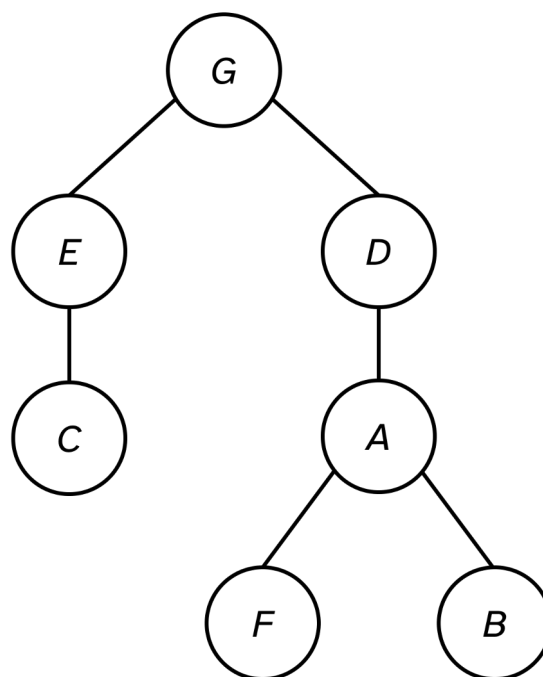
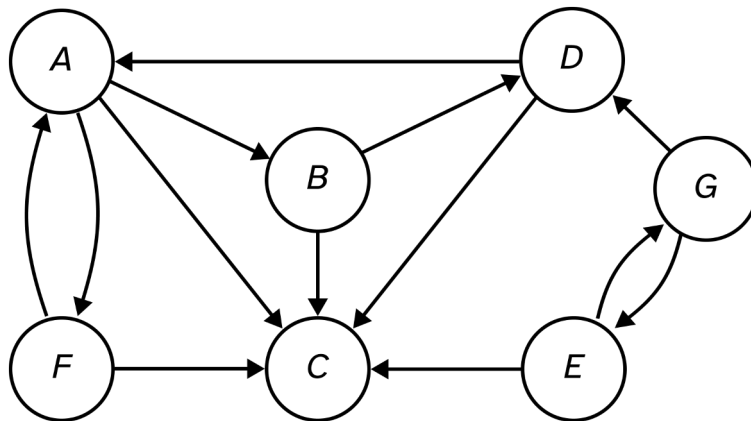


Without balancing, there would be two probes required to determine that 100 is not in the tree. The root, 6, would be checked and move to the right child 22. Since there is no right child to 22, we know that there exist no keys greater than 22 and can determine the key 100 is not present in the tree.

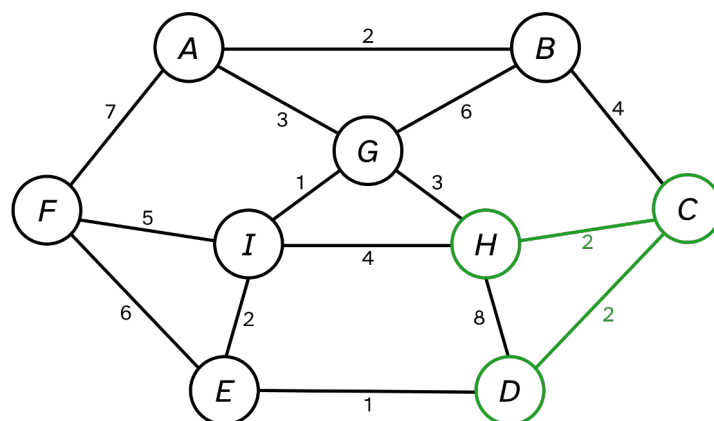
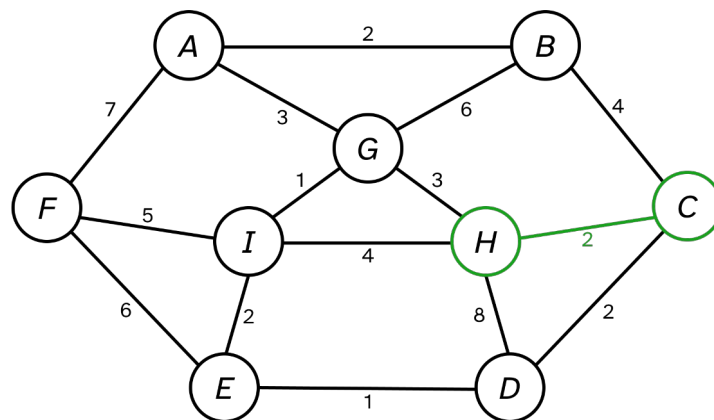
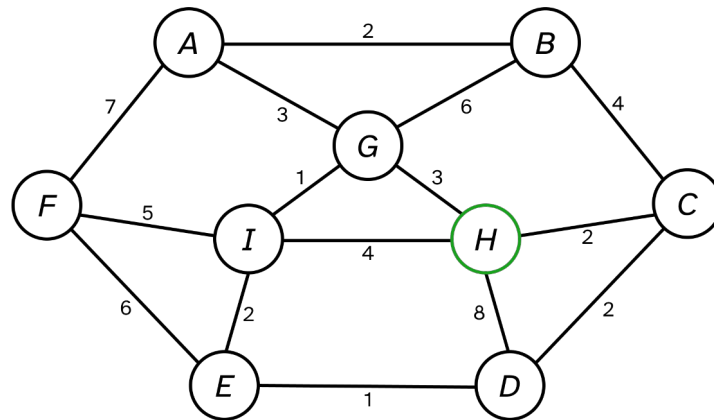
2. Find the depth-first search tree for the graph Fig. 1 with G as the starting vertex. Assume that each adjacency list is in alphabetical order.

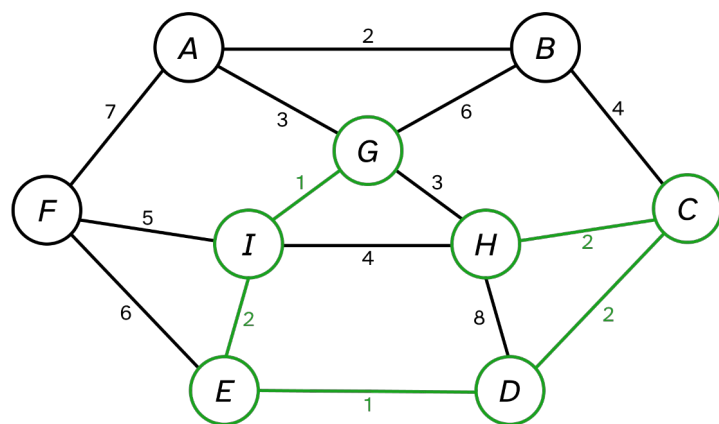
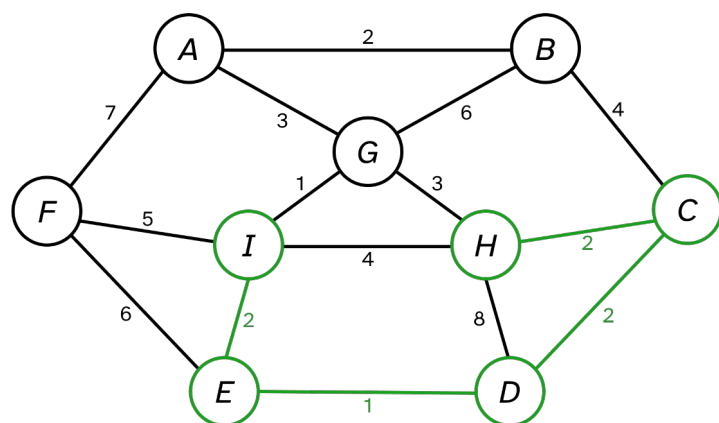
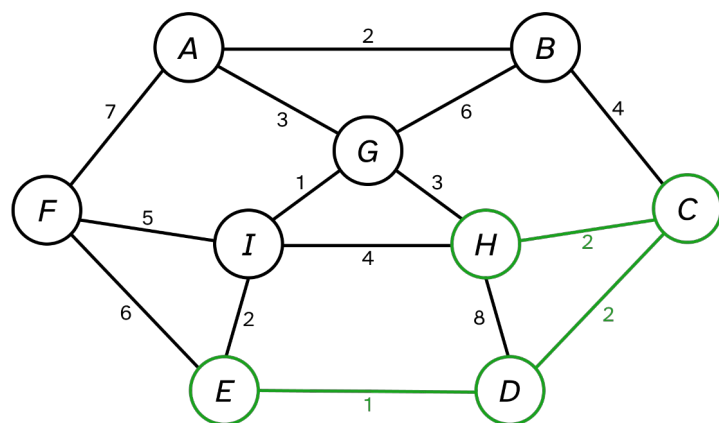


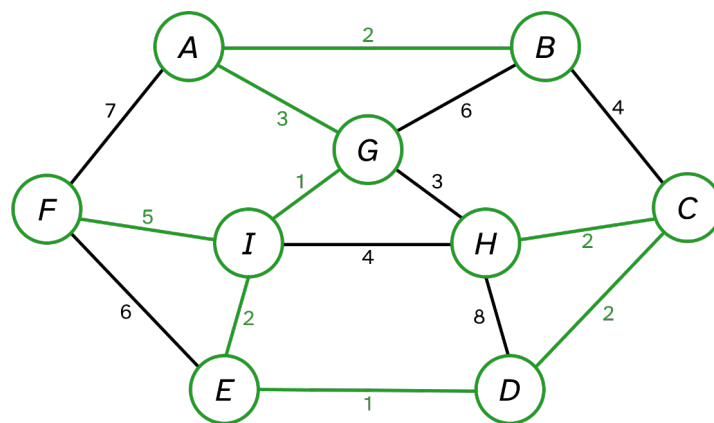
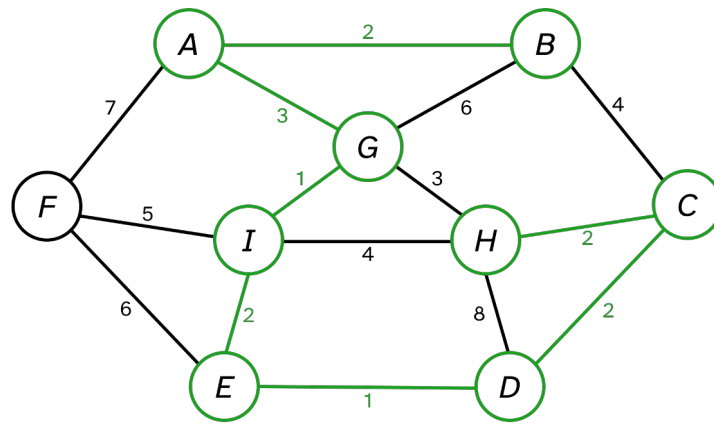
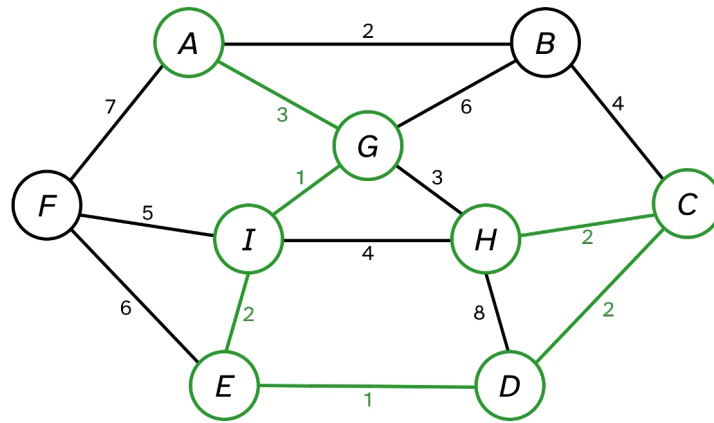
3. Find the breadth-first search tree for the graph Fig. 1 with G as the starting vertex. Assume that each adjacency list is in reverse alphabetical order.



4. Execute Prim's minimum spanning tree algorithm by hand on the graph in Fig. 2, showing how the data structures evolve. Clearly indicate which edges become part of the minimum spanning tree and in what order. Start at vertex *H*.



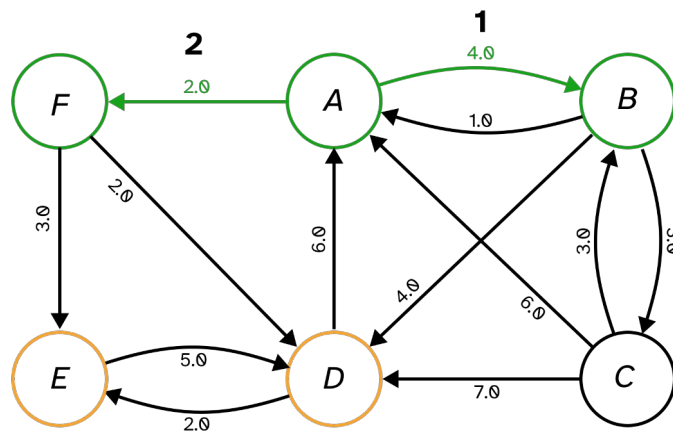
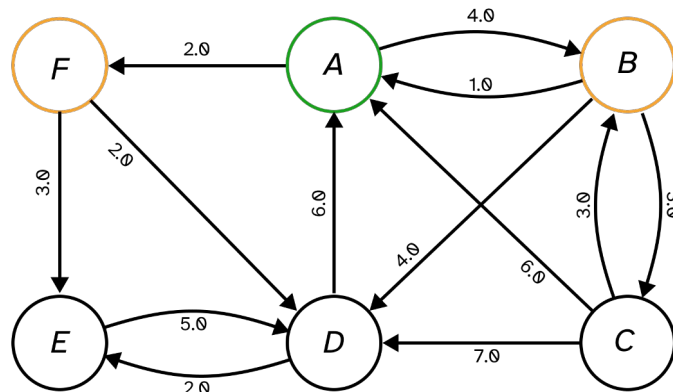


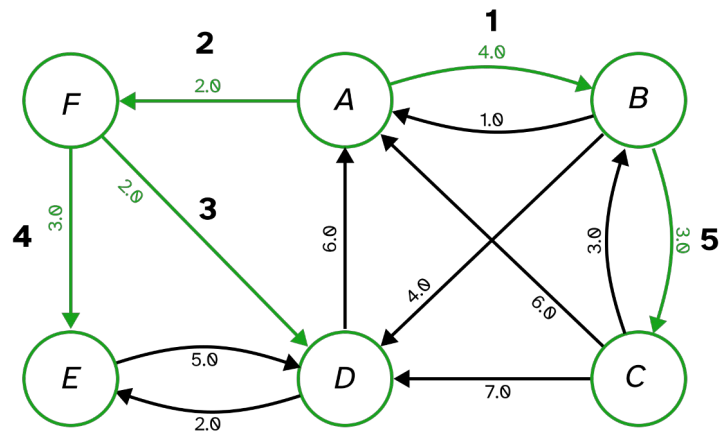
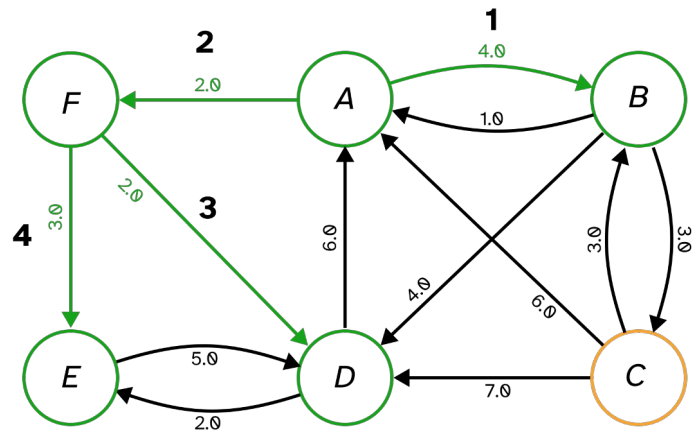


5. Dijkstra's shortest path algorithm. Here are the adjacency lists with edge weights in parentheses for a digraph Fig. 3.

A : B (4.0), F(2.0)
 B : A (1.0), C (3.0), D (4.0)
 C : A (6.0), B (3.0), D (7.0)
 D : A (6.0), E (2.0)
 E : D (5.0)
 F : D (2.0), E (3.0)

Execute Dijkstra's shortest-path algorithm by hand on this graph, showing how the data structures evolve, with $s = A$. Clearly indicate which edge becomes part of the shortest-path tree and in what order.





6. Find the minimum spanning tree for the graph in Fig. 4 that would be the output by Kruskal's algorithm, assuming that edges are sorted as shown on Fig. 4.

Sorted edges: *AB, EF, EK, FK, GH, GL, GM, HL, BC, CM, DJ, FG, JM, LM, AH, CD, CJ, HM, AI, AM, BI, DE, DM, IM, KM, BM, EM, FM*

