**CS330 Activities for Lecture 25\(^a\)**

### 10.5 Spanning Trees and Graph Traversals

- If \(G\) be a simple\(^1\), connected, undirected graph, then a **spanning tree** of \(G\) is any subgraph of \(G\) that is a tree that includes all the vertices of \(G\). A graph may have more than one spanning tree.
- We can build a spanning tree for \(G\) by traversing it, adding edges as we (unless they would introduce a circuit). Once all vertices are reached, the tree is complete.
- Two traversal algorithms are Depth-First Search and Breadth-First Search.
- Both operate by recursively visiting vertices and adding edges to already-unvisited vertices.
  - Say you visit vertex \(v\), and it has unmarked neighbors \(x_1, x_2, \ldots, x_n\).
  - In Depth-First search, you visit all the vertices reachable from \(x_1\) before you visit \(x_2\).
  - In Breadth-First search, you visit each of \(x_1, x_2, \ldots, x_n\) before going back and visiting the vertices reachable from \(x_1\) (and then from \(x_2\), etc).

**Questions**

1. Give depth-first and breadth-first traversals for the following graph. (Use alphabetical order to choose which adjacent vertices to visit first.)

![Graph Image]

2. Repeat #1 but swap the names of vertices a and e first.

---

* Some material from MATH230 Dr. Robert Ellis IIT-AMAT, zyBooks Disc. Math., and Disc. Math. and its Appl., Rosen (7th Ed.) is included under the academic/nonprofit Fair Use exception of US Copyright Law. This document must NOT be distributed beyond participants in IIT’s CS330.

\(^a\) Simple graph: No self-loops, no multiple edges between vertices.
4. A linear graph (one where the vertices all lie on one path) has the same DFS and BFS tree. Find one or more graphs with cycles that have the same DFS and BFS trees but where the root has more than one child.

5. Suppose you do a DFS on a graph and end up with a tree that doesn’t include all the nodes of the original graph. What does this say about the original graph?
6. Going back to the graph from problem 1, are there any edges that have to be included in any possible spanning tree, not just the trees found by DFS or BFS? Why? For a general graph, what kinds of edges have to appear in any possible spanning tree?

7. During DFS, if you come across an already-visited vertex, does that mean there's a cycle in the original graph? Or do you have to add any more assumptions to conclude that there's a cycle?

10.6 Minimum Spanning Trees

- If we have a simple, connected, undirected graph with weights attached to an edge, then a minimum spanning tree is a spanning tree whose total weight is as small as possible. (The total weight will be unique, though the tree might not be.)

Questions

8. Give an example of a graph with multiple minimum spanning trees.
Prim’s algorithm is a greedy algorithm for finding a minimum weight spanning tree. You start with an empty tree and add any arbitrary node of G to T. Now take all the edges with one vertex in T and the other vertex not in T and look at their weights. Choose an edge with the ones with the smallest weight (if there are ties, it doesn’t matter which one you pick) and add its other vertex to T. Repeat until there are no such edges left.

Questions

9. Use Prim’s algorithm on the graph below to get an MST. Is the MST unique for this graph?

10. Prim’s algorithm is greedy — it makes each decision based on local information. Yet it produces a globally optimal result. Any guesses as to why that is?