



Overview

This week is devoted to two classic graph problems that have found extensive applications for decades. While the basic algorithms are easy to understand, efficient implementations in practical situations typically require deployment of data structures such as the priority queue and union-find structures that we considered in Part I. We also consider the fact that simple variations can lead to deep and difficult computational problems.

Lecture 3: Minimum Spanning Trees. In this lecture we study the minimum spanning tree problem. We begin by considering a generic greedy algorithm for the problem. Next, we consider and implement two classic algorithm for the problem—Kruskal's algorithm and Prim's algorithm. We conclude with some applications and open problems.

Lecture 4: Shortest Paths. In this lecture we study shortest-paths problems. We begin by analyzing some basic properties of shortest paths and a generic algorithm for the problem. We introduce and analyze Dijkstra's algorithm for shortest-paths problems with nonnegative weights. Next, we consider an even faster algorithm for DAGs, which works even if the weights are negative. We conclude with the Bellman-Ford-Moore algorithm for edge-weighted digraphs with no negative cycles. We also consider applications ranging from content-aware fill to arbitrage.

To Do:

- **Programming Assignment: Content-Aware Resizing.** Implement a seam-carving algorithm for content-aware resizing.
- **Job Interview Questions.** Algorithmic interview questions based on the lecture material.
- **Suggested Readings.** Section 4.3 and 4.4 in *Algorithms, 4th edition*.

✓ Complete

Go to next item

