7.3 Kruskal’s Algorithm

- **Kruskal’s algorithm**: finds an MST of a weighted network; at each step, it chooses the cheapest available edge that does not form a circuit, 216
- **negative network**: the network obtained by changing the signs of the weights in the original network, 217
- **Kruskal’s algorithm for MaxSTs**: applies Kruskal’s algorithm for MSTs to the negative network, 218

## EXERCISES

### WALKING

#### 7.1 Networks and Trees

1. A computer lab has seven computers labeled A through G. The connections between computers are as follows:
   - A is connected to D and G
   - B is connected to C, E, and F
   - C is connected to B, E, and F
   - D is connected to B, E, and F
   - E is connected to B and C
   - F is connected to B and C
   - G is connected to A and D

   Is the lab set-up a computer network? Explain why or why not.

2. The following is a list of the electrical power lines connecting eight small towns labeled A through H.
   - A power line connecting A and D
   - A power line connecting B and C
   - A power line connecting B and E
   - A power line connecting C and G
   - A power line connecting D and F
   - A power line connecting D and H
   - A power line connecting E and G

   Do the power lines form a network? Explain why or why not.

3. Consider the network shown in Fig. 7-20.

   ![Figure 7-20](image)

   - (a) How many degrees of separation are there between D and J?
   - (b) How many degrees of separation are there between A and L?
   - (c) How many degrees of separation are there between A and K?

4. Consider the network shown in Fig. 7-21.

   ![Figure 7-21](image)

   - (a) How many degrees of separation are there between A and J?
   - (b) How many degrees of separation are there between E and L?
   - (c) How many degrees of separation are there between M and P?
   - (d) What is the largest degree of separation between a pair of vertices?

5. Consider the tree shown in Fig. 7-22 on the next page.

   ![Tree](image)
6. Consider the tree shown in Fig. 7-23.
   (a) How many degrees of separation are there between A and P?
   (b) How many degrees of separation are there between E and P?
   (c) How many degrees of separation are there between L and P?
   (d) What is the largest degree of separation between a pair of vertices?

7.2 Spanning Trees, MSTs, and MaxSTs

21. Consider the network shown in Fig. 7-24.
   (a) Find a spanning tree of the network.
   (b) Calculate the redundancy of the network.
   (c) What is the largest degree of separation between a pair of vertices in the network?

22. Consider the network shown in Fig. 7-25.
   (a) Find a spanning tree of the network.
   (b) Calculate the redundancy of the network.
   (c) What is the largest degree of separation between a pair of vertices in the network?

23. Consider the network shown in Fig. 7-26.
   (a) Find a spanning tree of the network.
   (b) Calculate the redundancy of the network.
   (c) What is the largest degree of separation between a pair of vertices in the network?
24. Consider the network shown in Fig. 7-27.
(a) Find a spanning tree of the network.
(b) Calculate the redundancy of the network.
(c) What is the largest degree of separation between a pair of vertices in the network?

25. (a) Find all the spanning trees of the network shown in Fig. 7-28(a).
(b) Find all the spanning trees of the network shown in Fig. 7-28(b).
(c) How many different spanning trees does the network shown in Fig. 7-28(c) have?

26. (a) Find all the spanning trees of the network shown in Fig. 7-29(a).
(b) Find all the spanning trees of the network shown in Fig. 7-29(b).
(c) How many different spanning trees does the network shown in Fig. 7-29(c) have?

27. (a) How many different spanning trees does the network shown in Fig. 7-30(a) have?
(b) How many different spanning trees does the network shown in Fig. 7-30(b) have?

28. (a) How many different spanning trees does the network shown in Fig. 7-31(a) have?
(b) How many different spanning trees does the network shown in Fig. 7-31(b) have?

29. Consider the network shown in Fig. 7-32.
(a) How many different spanning trees does this network have?
(b) Find the spanning tree that has the largest degree of separation between $H$ and $G$.
(c) Find a spanning tree that has the smallest degree of separation between $H$ and $G$. 

FIGURE 7-26

FIGURE 7-27

FIGURE 7-28

FIGURE 7-29

FIGURE 7-30

FIGURE 7-31

FIGURE 7-32
49. Suppose that in a weighted network there is just one edge (call it XY) with the smallest weight. Explain why the edge XY must be in every MST of the network.

50. Suppose that in a weighted network there is just one edge (call it XY) with the largest weight.
   (a) Give an example of a network with more than one MST and such that XY must be in every MST.
   (b) Give an example of a network with more than one MST and such that XY is in none of the MSTs.

51. Suppose G is a disconnected graph with N vertices, M edges, and no circuits.
   (a) How many components does the graph have when N = 9 and M = 6?
   (b) How many components does the graph have when N = 240 and M = 236? Explain your answer.

52. Suppose G is a disconnected graph with no circuits. Let N denote the number of vertices, M the number of edges, and K the number of components. Explain why M = N - K. (Hint: Try Exercise 51 first.)

53. Cayley's theorem. Cayley's theorem says that the number of spanning trees in a complete graph with N vertices is given by N^{N-2}.
   (a) List the 4^2 = 16 spanning trees of K_4.
   (b) Which is larger, the number of Hamilton circuits or the number of spanning trees in a complete graph with N vertices? Explain.

PROJECTS AND PAPERS

1 Other Algorithms for MSTs and MaxSTs
Kruskal's algorithm is not the only good algorithm for finding MSTs and MaxSTs in a weighted network—two other well-known algorithms that are used to solve the same problem are Prim's algorithm and Boruvka's algorithm. Prepare a presentation on Prim's and Boruvka's algorithms. Describe both algorithms, and demonstrate how they work using one of the examples from the chapter. Compare Prim's and Boruvka's algorithms to Kruskal's algorithm, and discuss their similarities and differences.

2 Dijkstra's Shortest-Path Algorithm
In a network, there is at least one, but often many, paths connecting two vertices X and Y. When the network is an ordinary network, the "shortest path" between X and Y is defined as the path with the fewest number of edges (and the number of edges in a shortest path is called the degree of separation between X and Y). On the other hand, when the network is a weighted network the "shortest path" between X and Y has a different definition: Among all paths connecting X and Y, it is the path of least total weight.

Finding the shortest path between pairs of vertices in a weighted network is a problem that has many similarities to the problem of finding a minimum spanning tree of the network, and there is a nice algorithm that solves the problem optimally and efficiently. The algorithm, known as Dijkstra's algorithm, is named after the Dutch computer scientist Edsger Dijkstra, who first proposed the algorithm in 1959. In this project you are asked to prepare a presentation on Dijkstra's algorithm. At the very least you should carefully describe the algorithm, discuss some of its applications to real-world problems, and illustrate the algorithm with a couple of examples. Dijkstra's algorithm is a well-known algorithm, and you will find plenty of resources for this project on the Web.

3 Social Networks and Privacy
There is an inverse relation between connectedness and privacy. In general, the more connected you are the less privacy you have. The problem gets worse when it comes to