Assignment #14

set 23.2 : Problems 2,4

set 23.3 : Problem 4

set 23.4 : Problem 6

set 23.5: Problem 6

set 23.7 : Problems 8,20

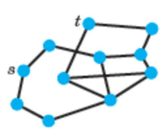
PROBLEM SET 23.2

SHORTEST PATHS, MOORE'S BFS

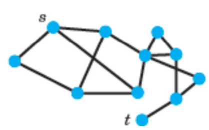
(All edges length one)

1–4 Find a shortest path $P: s \to t$ and its length by Moore's algorithm. Sketch the graph with the labels and indicate P by heavier lines as in Fig. 482.

1.



2.



3.

4.

5. Moore's algorithm. Show that if vertex v has label $\lambda(v) = k$, then there is a path $s \to v$ of length k.

6. Maximum length. What is the maximum number of edges that a shortest path between any two vertices in a graph with n vertices can have? Give a reason. In a complete graph with all edges of length 1?

PROBLEM SET 23.3

1. The net of roads in Fig. 488 connecting four villages is to be reduced to minimum length, but so that one can still reach every village from every other village. Which of the roads should be retained? Find the solution **(a)** by inspection, **(b)** by Dijkstra's algorithm.

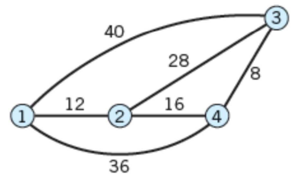


Fig. 488. Problem 1

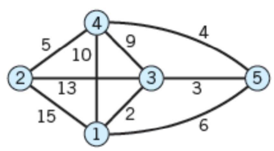
2. Show that in Dijkstra's algorithm, for L_k there is a path P: $1 \rightarrow k$ of length L_k.

3. Show that in Dijkstra's algorithm, at each instant the demand on storage is light (data for fewer than n edges).

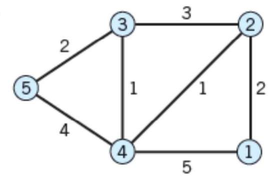
4–9 **DIJKSTRA'S ALGORITHM**

For each graph find the shortest paths.

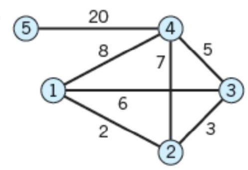
4.



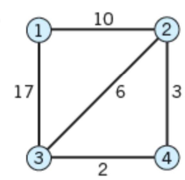
5.



6.



7.

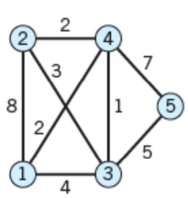


PROBLEM SET 23.4

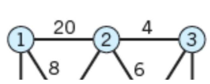
1–6 **KRUSKAL'S GREEDY ALGORITHM**

Find a shortest spanning tree by Kruskal's algorithm. Sketch it.

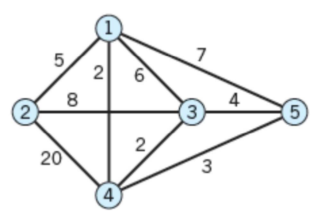
1.



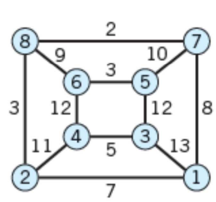
2.



5.



6.



PROBLEM SET 23.5

SHORTEST SPANNING TREES. PRIM'S ALGORITHM

1. When will $S = E$ at the end in Prim's algorithm?

2. **Complexity.** Show that Prim's algorithm has complexity $O(n^2)$.

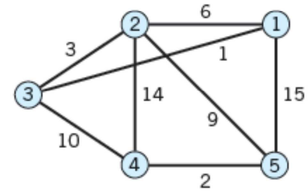
3. What is the result of applying Prim's algorithm to a graph that is not connected?

4. If for a complete graph (or one with very few edges missing), our data is an $n \times n$ distance table (as in Prob. 13, Sec. 23.4), show that the present algorithm [which is $O(n^2)$] cannot easily be replaced by an algorithm of order less than $O(n^2)$.

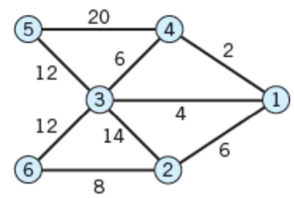
5. How does Prim's algorithm prevent the generation of cycles as you grow T?

6–13 Find a shortest spanning tree by Prim's algorithm.

6.



7.



PROBLEM SET 23.6

1–6 **CUT SETS, CAPACITY**

Find T and cap (S, T) for:

1. Fig. 498, $S = \{1, 2, 4, 5\}$
2. Fig. 499, $S = \{1, 2, 3\}$
3. Fig. 498, $S = \{1, 2, 3\}$
4. Fig. 499, $S = \{1, 2\}$
5. Fig. 499, $S = \{1, 2, 4, 5\}$
6. Fig. 498, $S = \{1, 3, 5\}$

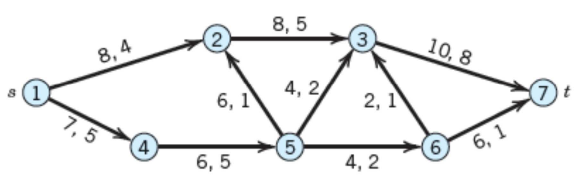
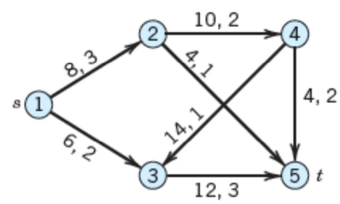
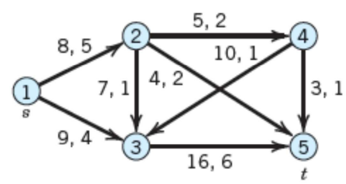


Fig. 499. Problems 2, 4, and 5

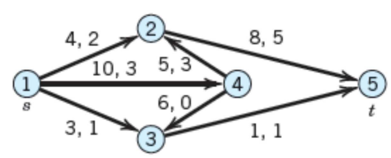
13.



14.



15.



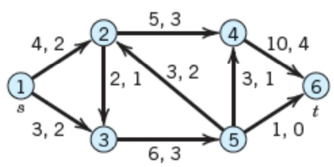
PROBLEM SET 23.7

1. Do the computations indicated near the end of Example 1 in detail.
2. Solve Example 1 by Ford–Fulkerson with initial flow 0. Is it more work than in Example 1?
3. Which are the "bottleneck" edges by which the flow in Example 1 is actually limited? Hence which capacities could be decreased without decreasing the maximum flow?
4. What is the (simple) reason that Kirchhoff's law is preserved in augmenting a flow by the use of a flow augmenting path?
5. How does Ford–Fulkerson prevent the formation of cycles?

6–9 MAXIMUM FLOW

Find the maximum flow by Ford-Fulkerson:

6. In Prob. 12, Sec. 23.6
7. In Prob. 15, Sec. 23.6
8. In Prob. 14, Sec. 23.6
9.



10. **Integer flow theorem.** Prove that, if the capacities in a network G are integers, then a maximum flow exists and is an integer.
11. **CAS PROBLEM. Ford–Fulkerson.** Write a program and apply it to Probs. 6–9.
12. How can you see that Ford–Fulkerson follows a BFS technique?
13. Are the consecutive flow augmenting paths produced by Ford–Fulkerson unique?
14. If the Ford–Fulkerson algorithm stops without reaching t, show that the edges with one end labeled and the other end unlabeled form a cut set (S, T) whose capacity equals the maximum flow.
15. Find a minimum cut set in Fig. 500 and its capacity.
16. Show that in a network G with all $c_{ij} = 1$, the maximum flow equals the number of edge-disjoint paths $s \to t$.
17. In Prob. 15, the cut set contains precisely all forward edges used to capacity by the maximum flow (Fig. 501). Is this just by chance?
18. Show that in a network G with capacities all equal to 1, the capacity of a minimum cut set (S, T) equals the minimum number q of edges whose deletion destroys all directed paths $s \to t$. (A **directed path** $v \to w$ is a path in which each edge has the direction in which it is traversed in going from v to w.)

19. **Several sources and sinks.** If a network has several sources $s_1, \cdots, s_k$, show that it can be reduced to the case of a single-source network by introducing a new vertex s and connecting s to $s_1, \cdots, s_k$ by k edges of capacity ∞. Similarly if there are several sinks. Illustrate this idea by a network with two sources and two sinks.
20. Find the maximum flow in the network in Fig. 502 with two sources (factories) and two sinks (consumers).

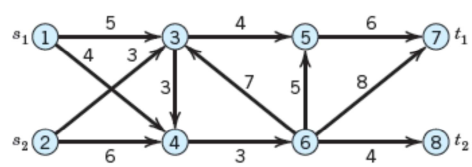


Fig. 502. Problem 20