COMP 3270 Homework Assignment 4
Seth Denney

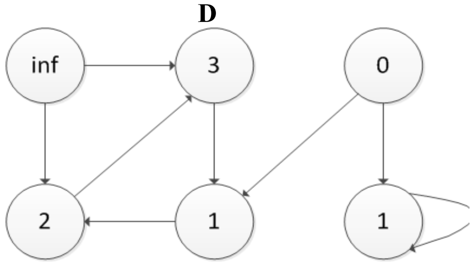
Feel free to discuss this assignment with others; however, all work should reflect your own understanding and you should never copy another's work. One useful rule of thumb is, "would I be able to explain my answer to someone else?"

Problems
Chap 22

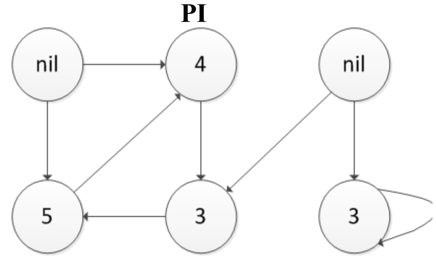
Exercise 22.1-1 . Given an adjacency-list representation of a directed graph, how long does it take to compute the out-degree of every vertex? How long does it take to compute the in-degrees?
*The computation of the out-degree of a vertex of a directed graph in an adjacency list representation would consist of measuring the length of the linked list at that particular vertex's array index. This could be a linear search along the linked list, or, if there is a "size" field stored, it could be constant time for each index, resulting in O(V) time for all vertices. The time to compute the in-degree of each vertex will require a search of ALL linked lists in the adjacency list, which means that computing the in-degree of all vertices would take O(V*E) time.*

Exercise 22.2-1 Show the d and pi values that result from running breadth-first search on the directed graph of Fig 22.2(a) using vertex 3 as the source.

D



PI

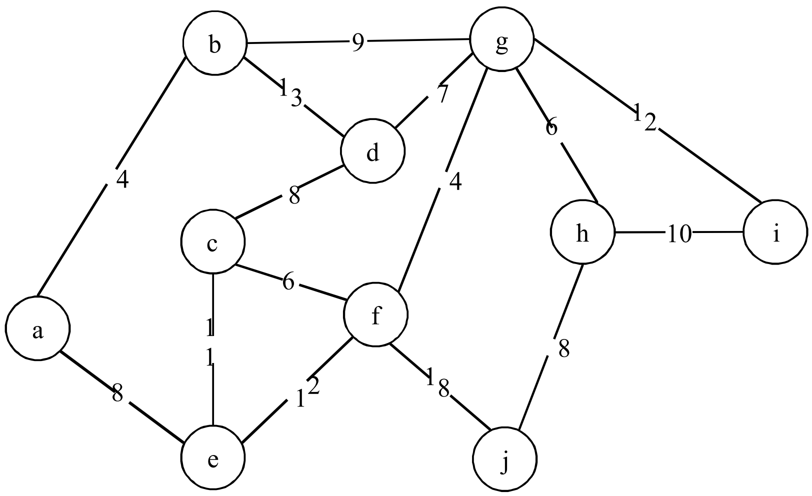


Exercise 22-4-1 Show the ordering of vertices produced by TOPOLOGICAL-SORT when it is run on the dag of Figure 22.8 under the assumption of Exercise 22,3,2, ie vertices are considered in alphabetical order.

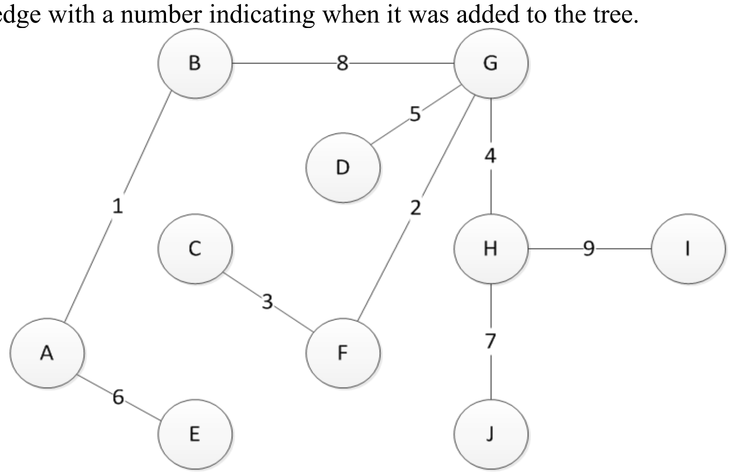
	START	FINISH
M	1	20
N	21	26
O	22	25
P	27	28
Q	18	19
R	2	17
S	23	24
T	4	5
U	3	6
V	8	15
W	9	12
X	13	14
Y	7	16
Z	10	11

28	26	25	24	20	19	17	16	15	14	12	11	6	5
P	**N**	**O**	**S**	**M**	**Q**	**R**	**Y**	**V**	**X**	**W**	**Z**	**T**	**U**

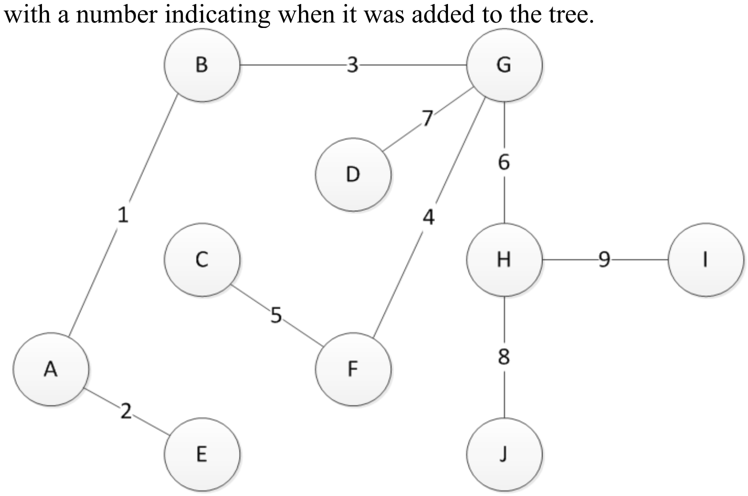
Chap 23



1. Show the minimum spanning tree generated by Kruskal's algorithm on the graph above. Label each tree edge with a number indicating when it was added to the tree.

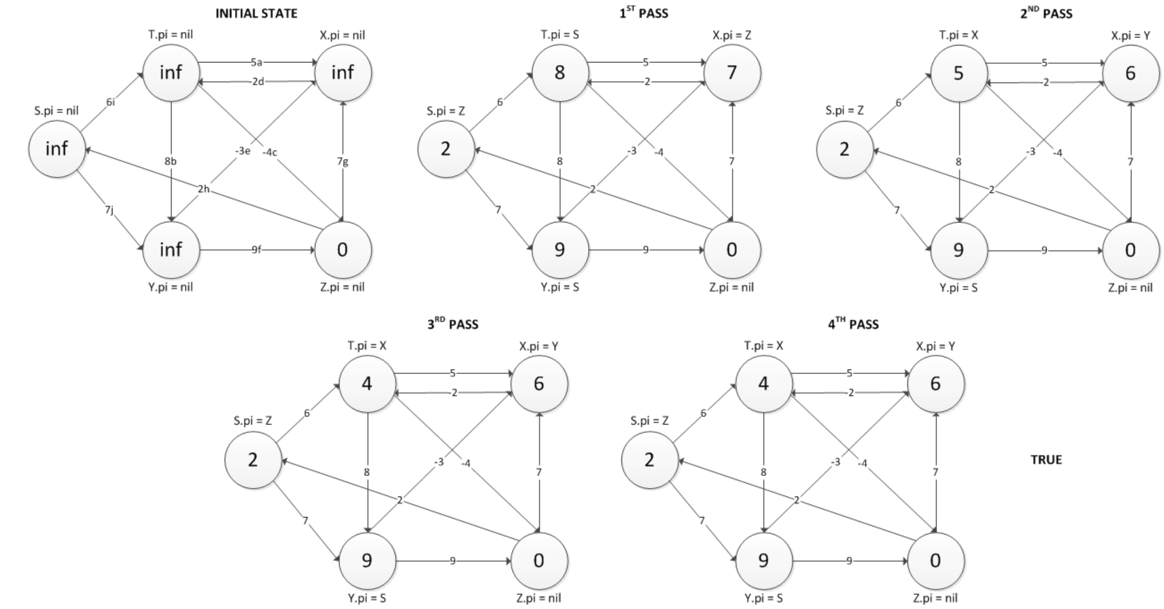


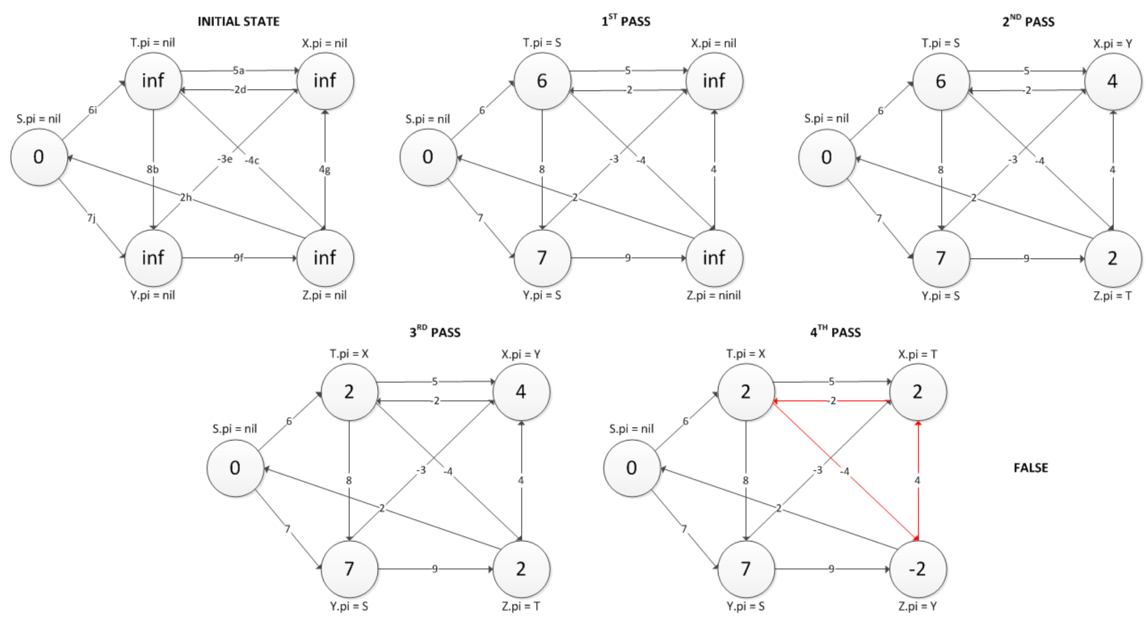
2. Show the minimum spanning tree generated by Prim's algorithm on the graph above. Label each tree edge with a number indicating when it was added to the tree.



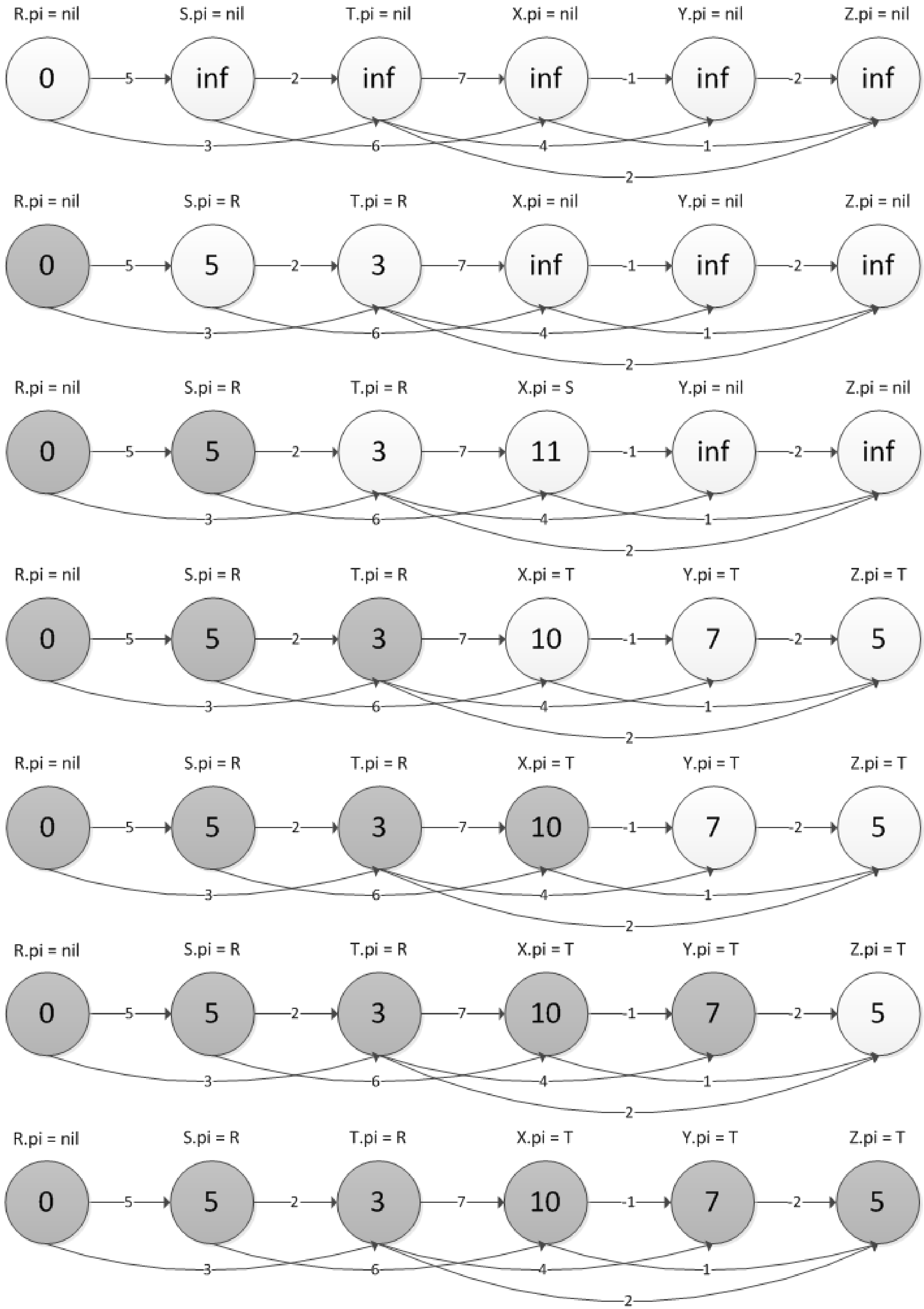
Chap 24

Exercise 24.1-1 Run the Bellman-Ford algorithm on the directed graph of Figure 24.2, using vertex z as the source. In each pass, relax edges in the same order as in the figure and show the d and pi values after each pass. Now, change the weight of edge (z,x) to 4 and run the algorithm again using s as the source.

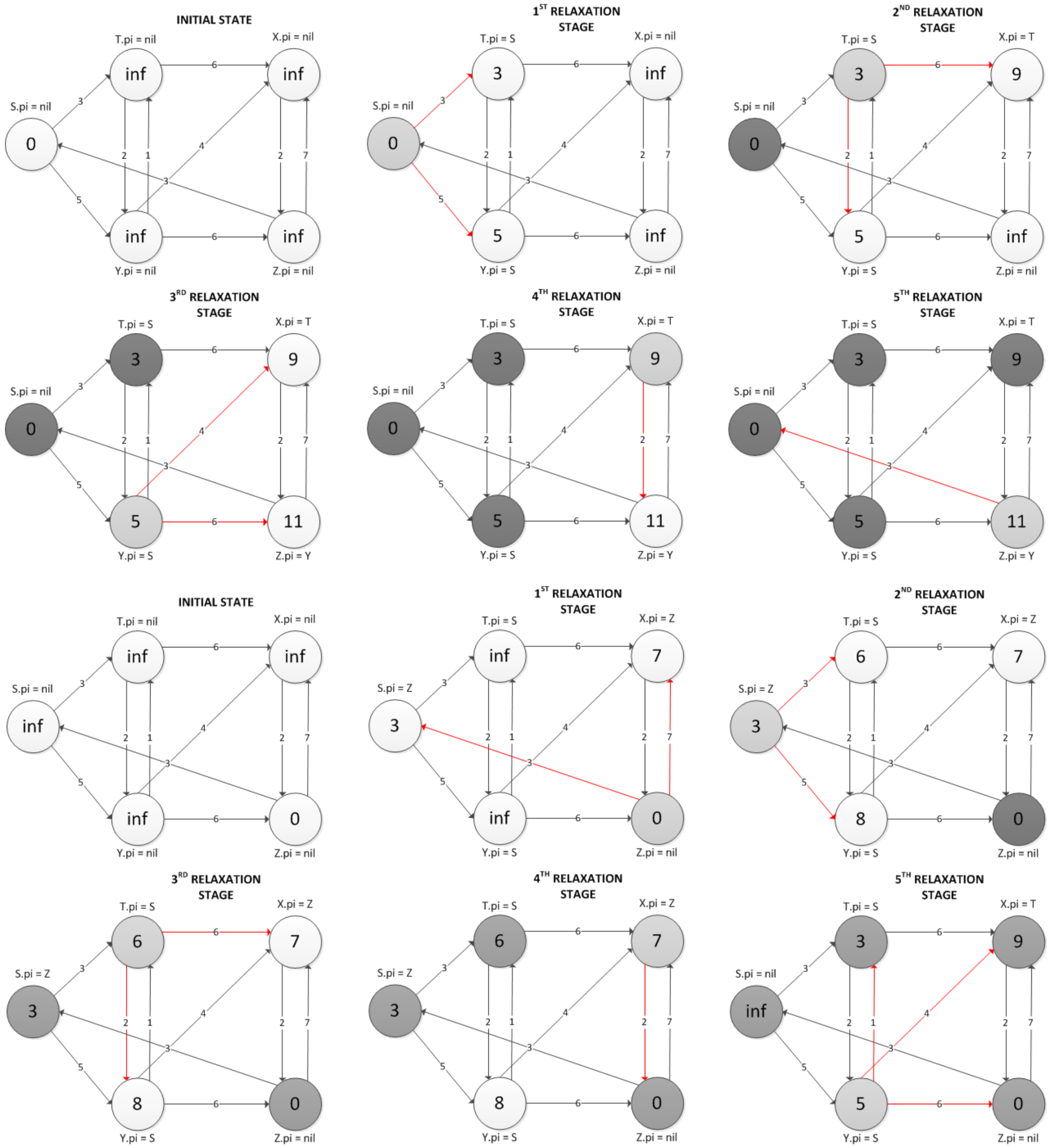




Exercise 24.2-1 Run DAG_SHORTEST_PATHS on the directed graph of Figure 24.5 using vertex r as the source.

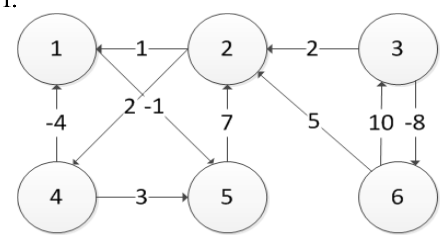


Exercise 24.3-1 Run Dijkstra's algorithm on the digraph of Figure 24.2, using vertex s as the source and then using vertex z as the source. In the style of Figure 24.6, show the d and pi values an the vertices in the Set S after each iteration of the while loop.



Chap 25

Exercise 25.1-1 Run SLOW-ALL-PAIRS-SHORTEST-PATH on the weighted, directed graph of Figure 25.2, showing the matrices that result for each iteration of the loop. Then do the same for FASTER-ALL-PAIRS-SHORTEST-PATH.



L^1 SLOW-ALL-PAIRS-SHORTEST-PATH

0	inf	inf	inf	-1	inf
1	0	inf	2	inf	inf
inf	2	0	inf	inf	-8
-4	inf	inf	0	3	inf
inf	7	inf	inf	0	inf
inf	5	10	inf	inf	0

L^2

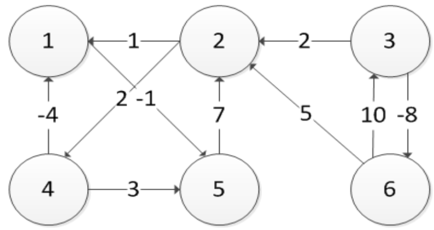
0	6	inf	inf	-1	inf
-2	0	inf	2	0	inf
3	-3	0	4	inf	-8
-4	10	inf	0	-5	inf
8	7	inf	9	0	inf
6	5	10	7	inf	0

L^3

0	6	inf	8	-1	inf
-2	0	inf	2	-3	inf
-2	-3	0	-2	2	-8
-4	2	inf	0	-5	inf
5	7	inf	9	0	inf
3	5	10	7	10	0

$L^4 = L^5$

0	6	inf	8	-1	inf
-2	0	inf	2	-3	inf
-5	-3	0	-1	-3	-8
-4	2	inf	0	-5	inf
5	7	inf	9	0	inf
3	5	10	7	2	0



L^1 FASTER-ALL-PAIRS-SHORTEST-PATH

0	inf	inf	inf	-1	inf
1	0	inf	2	inf	inf
inf	2	0	inf	inf	-8
-4	inf	inf	0	3	inf
inf	7	inf	inf	0	inf
inf	5	10	inf	inf	0

L^2

0	6	inf	inf	-1	inf
-2	0	inf	2	0	inf
3	-3	0	4	inf	-8
-4	10	inf	0	-5	inf
8	7	inf	9	0	inf
6	5	10	7	inf	0

$L^4 = L^5$

0	6	inf	8	-1	inf
-2	0	inf	2	-3	inf
-5	-3	0	-1	-3	-8
-4	2	inf	0	-5	inf
5	7	inf	9	0	inf
3	5	10	7	2	0

Chap 26

Exercise 26.2-2 In Figure 26.1(b) what is the flow across the cut ({s,v2,v4}, {v1,v3,t}) ? What is the capacity of this cut?

The flow across the cut, f = 11 + 1 + 7 + 4 – 4 = 19.
The capacity of the cut, c = 16 + 4 + 7 + 4 = 31.

Chap 34

Exercise 34.4-6 Suppose someone gives you a polynomial-time algorithm to decide formula satisfiability. Describe how to use this algorithm to find satisfying assignments in polynomial time.

A formula is satisfiable if it can be shown that there exists some combination of assignments to the boolean arguments $x_1, \dots, x_m$ for which the boolean formula evaluates to *true*. If a polynomial-time algorithm to decide formula satisfiability is given, then said algorithm must provide a mechanism for finding such a combination of assignments in polynomial time. If the output of the algorithm is modified so that the algorithm no longer acts as a decision algorithm (simply returning *true* or *false*), and instead returns the satisfying assignment that was found for the given boolean formula (if any), then the algorithm can be used to find satisfying assignments in polynomial time.