Solutions

Student Name and ID Number

MATH 3012, Quiz 3, April 19, 2013, WTT

1. A single die (six sides) is rolled. If the result is a one, two or three, the payoff is \$4. If the result is a four or five, then the payoff is \$7, and if the result is a six, then the payoff is \$10. What is the expected value of the payoff?

5

$$\frac{3}{6}\cdot 4 + \frac{2}{6}\cdot 7 + \frac{1}{6}\cdot 10$$

2. A single die is rolled as many times as it takes to get a result which is *not* a six. When this happens, you win if the result is a one, two or three and lose if it is a four or five. What is the probability of winning?

5

$$\frac{\frac{3}{6}}{\frac{3}{6}+\frac{2}{6}} = \frac{3}{5}$$

3. Consider the data file (shown on the left below) for the weights on the edges of a graph with vertex set $\{a, b, c, d, e, f, g, h\}$. In the space to the right, list in order the edges that would be selected in carrying out Kruskal's algorithm (avoid cycles) and Prim's algorithm (build tree) to find a minimum weight spanning tree. For Prim, use vertex a as the root.

20

10 points each

graph-data.txt

be	8
df	9
ce	10
bc	11
dh	12
fh	13
he	14
gh	15
bh	16
fg	16
ad	17
af	18

Kruskal

be 8
df 9
ce 10
dh 12
~~fh 13~~
~~gh 15~~
~~bh 16~~
~~ad 17~~
he 14
gh 15
ad 17

Prim

ad 17
df 9
dh 12
~~fh 13~~
eh 14
eb 8
ec 10
gh 15

21

4. A data file `digraph_data.txt` has been read for a digraph whose vertex set is $\{1, 2, 3, 4, 5, 6, 7\}$. The weights on the directed edges are shown in the matrix below. The entry $w(i, j)$ denotes the length of the edge from i to j. If there is no entry, then the edge is not present in the graph. In the space to the right, apply Dijkstra's algorithm to find the distance from vertex 1 to all other vertices in the graph. Also, for each x, find a shortest path from 1 to x.

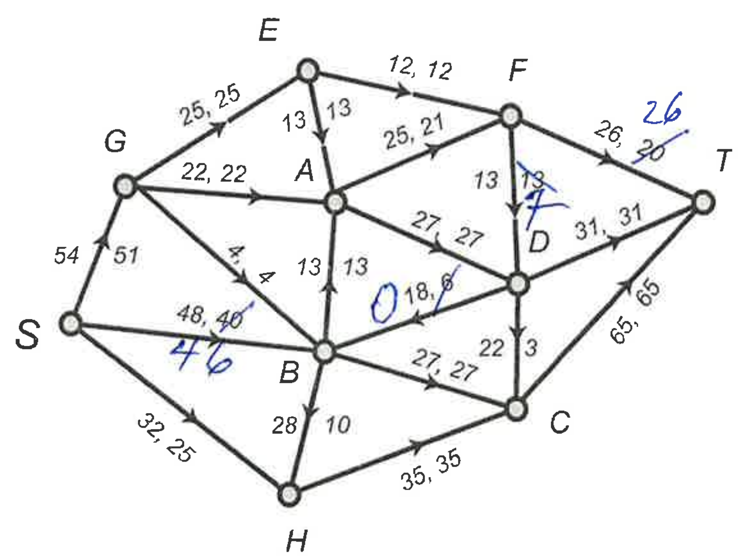
W	1	2	3	4	5	6	7
1	0		10		45	8	
2		0		9			
3			0	16	34		18
4		20		0			
5					0		
6		7		18		0	
7							0

2	3	4	5	6	7
(1,2) ∞	(1,3) 10	(1,4) ∞	(1,5) 45	$\boxed{(1,6)\ 8}$	(1,7) ∞
(1,6,2) 15	$\boxed{(1,3)\ 10}$	(1,6,4) 26	(1,5) 45		(1,7) ∞
$\boxed{(1,6,2)\ 15}$		(1,6,4) 26	(1,3,5) 44		(1,3,7) 28
		$\boxed{(1,6,2,4)\ 24}$	(1,3,5) 44		(1,3,7) 28
			(1,3,5) 44		$\boxed{(1,3,7)\ 28}$
			$\boxed{(1,3,5)\ 44}$		

5. Consider the following network

24

3 points each part



a. What is the current value of the flow?

51 + 40 + 25 = 116

b. What is the capacity of the cut $\{S, A, B, E, G, H\} \cup \{C, D, F, T\}$?

12 + 25 + 27 + 27 + 35

c. Write below the labels that are applied by carrying out the Ford-Fulkerson labeling algorithm, using the "pseudo-alphabetic" ordering on nodes.

S (*, +, ∞)
B (S, +, 8)
G (S, +, 3)
H (S, +, 7)
D (B, −, 6)
A (D, −, 6)
C (D, +, 6)
~~E (A, −, 6)~~
E (A, ■, 6)
T (F, +, 6)

d. Write the sequence of vertices that forms an augmenting path, as determined by the labeling done in the previous step.

(S, B, D F, T)

e. Use the information gleaned from the previous two parts to update the flow. You may provide your answer by writing directly on the figure.

f. What is the new value value of the flow?

$116 + 6 = 122$

g. Write below the labels that are applied by carrying out the Ford-Fulkerson labeling algorithm on the updated network. It should terminate without the sink being labeled.

S (*, +, ∞)
B (S, +, 2)
G (S, +, 3)
H (S, +, 7)

h. Find a cut whose capacity is the value of the current flow.

S, B, G, H || A, C, D, E, F, T

15

5 points each part

6. Let P be a poset whose ground set is $\{a, b, c, d, e, f, g, h\}$. A bipartite matching algorithm is used to solve the Dilworth problem for this poset. When the algorithm halts, the edges in the maximum matching are listed below on the left.

a. In the space to the right, assemble the chain partition associated with this matching.

$d'c''$		f	
$b'a''$	c	a	h
$a'f''$	d	b	
$g'd''$	g	e	
$e'b''$			

b. What is the width of the poset P.

3

c. Explain how you would use the labelling information obtained when the network flow (bipartite matching) algorithm halts to find a maximum antichain in P.

For each chain C in the partition, we choose a point x from C so that x' is labelled and x'' is unlabelled. These points form an antichain in P

10

2 points each

7. True–False. Mark in the left margin.

F 1. Linear programming problems with integer coefficient constraints always have integer valued solutions.

T 2. The Ford-Fulkerson labelling algorithm uses an embedded instance of Dijkstra's algorithm to find an augmenting path with the fewest number of edges.

F 3. At each iteration, the Ford-Fulkerson algorithm will choose an augmenting path resulting in the maximum increase in the flow.

F 4. Let $G(X, Y)$ be a bipartite graph with $|X| = 400$, $|Y| = 600$. Then $G(X, Y)$ always has a matching of size 350.

F 5. When P is a poset on 100 points and the width of P is 12, there is always a partition of P into 12 antichains.

Fun 6. When a saturated tri-partite network is flowing into integer valued components, an effective algorithm for merging the advancement operators into Dilworth chains will always converge.

(Not graded)