MATH 3330 — Applied Graph Theory
Assignment 3 – Solutions

1. Give the number of 4-cycles in the hypercubes Q_2 and Q_3. Using the iterative way of constructing Q_n, derive a formula for the number of 4-cycles in Q_4. Finally, give a general formula for the number of 4-cycles in Q_n, for all $n \geq 2$.

   Let c_n be the number of 4-cycles in Q_n. By inspection, $c_2 = 1$ and $c_3 = 6$. As shown in class, Q_{n+1} can be formed by taking two copies of Q_n, and joining all corresponding vertices. Each 4-cycle in Q_n thus gives two 4-cycles in Q_{n+1}, one for each copy. In addition, there are the 4-cycles that contain vertices from both copies. Since each vertex is adjacent to exactly one vertex in the other copy, the only way this can happen is by taking two adjacent vertices in one copy, and the corresponding vertices in the other copy. Therefore, every edge in Q_n leads to one "cross-copy" 4-cycle in Q_{n+1}. We saw in class (or can be obtained using the degree-sum formula) that the number of edges in Q_n equals $n2^{n-1}$. Therefore, we get the recursive formula:

   $$c_{n+1} = 2c_n + n2^{n-1} = 2^2 c_{n-1} + 2(n-1)2^{n-2} + n2^{n-1} = ...$$

   This leads to the general formula:

   $$c_n = (1 + 2 + \ldots + (n-1))2^{n-2} = n(n-1)2^{n-3}.$$

   (2 points, last part bonus)

2. Suppose that $G = (V, E)$ is the intersection graph of a collection of sets S_v, which are such that the union of all the S_v contain only four elements. (So $|\bigcup_{v \in V} S_v| = 4$.)

   (a) Can G contain an induced 4-cycle? *Yes. For example, take the sets $\{a, b\}$, $\{b, c\}$, $\{c, d\}$, $\{d, a\}$.*

   (b) Can G contain an induced 5-cycle? *No. Note first that every set associated with vertices of the cycle must have at least 2 elements in it: if a set has only 1 element, then both of its neighbours must contain that element, which would mean that those neighbours are connected, and the 5-cycle has a chord. By the same reasoning, any element can contain in at most 2 neighbouring vertices. Since* *there are 5 vertices, each with sets of size at least 2, and we can use each element in at most 2 vertex sets, there must be a total of at least 5 elements. 2 points.*

3. A telephone company has to decide which of its lines to use as backbones, to route long distance traffic. It wants to connect n relay stations, which already have lines connecting them. For each existing line ℓ_{ij} connecting stations i and j, the probability of failure is known. The probability of failure of the lines are considered independent, so the probability that, at any time, two lines ℓ_{ij} and ℓ_{km} are both not failing equals the product $(1 - p_{ij})(1 - p_{km})$. The company wants to choose a backbone network that will form a *spanning tree* connecting all n stations, but so that the probability of the network failing is minimized (the network fails if *any* of its lines fails, because then the backbone is disconnected.)

   (a) An engineer with the company explains that this problem is equivalent to the minimum spanning tree problem, if the weight of each edge ℓ_{ij} is taken to be $-\log(1 - p_{ij})i$. Explain why the engineer is correct.
   Note that the non-failure probability of a given spanning tree equals the **product** *of the non-failure probabilities (i.e. probabilities $(1 - p_{ij})$ for each edge ij that is part of the tree) of the edges of the tree. So the goal is to find a spanning tree for which the product of the edge weights $(1 - p_{ij})$ is maximized. However, the logarithm of a product is the sum of the logarithms of the terms. Also, logarithm is an increasing function, so whatever maximizes the log, also maximizes the orginal quantity. By taking the negative of the logarithm, we change a maximum into a minimum. So the spanning tree for which the product of the weights $(1 - p_{ij})$ is maximized is the same as the spanning tree for which the sum of modified weights $-\log(1 - p_{ij})$ is miminized.*

   (b) The engineer proposes the following algorithm: Order all lines in order of increasing failure probability (so the first line will be the one with lowest failure probability, etc.). Then, choose lines in order. If a line does not form a cycle with lines already chosen, add it to the spanning tree. Otherwise, ignore it. Give a reasoning that this algorithm is equivalent to Kruskal's algorithm applied to

the weights described in (a).
This follows immediately from the argument given in (a). Kruskal's algorithm says to select, at each point, the edge with minimum weight that does not create a cycle. But the edge with minimum value $-\log(1-p_{ij})$ *is exactly the edge for which* $(1-p_{ij})$ *is highest, so* p_{ij} *is lowest.*

(c) Demonstrate the algorithm on a network of 5 stations labelled 1,2,3,4,5. The failure probabilities are given in the table below.

	2	3	4	5
1	0.01	0.01	0.02	0.005
2		0.015	0.017	0.005
3			0.017	0.02
4				0.007

Choose edges 1–5, 2–5, 4–5, (skip 1–2 since it creates a cycle), 1–3. Total: 2 points. Sum/product conversion should be mentioned to get full marks.

4. Give a detailed argument why Kruskal's algorithm cannot be used to find a minimum directed spanning tree in a strongly connected graph.

   Every student in the class gave a valid argument. Basically, Kruskal's could fail for two reasons. Either you get a more expensive tree because you optimize locally which may lead to a more expensive edge at the end, or you fail to get a rooted tree all-together, because Kruskal's allows for disconnected pieces which will form a tree in the end. However, in the directed tree it is not always the case that the disconnected pieces will give you the directed, rooted tree you want. 1 point.

5. Do problems 3.1.6–3.1.8 of the text.

   3.1.6. Impossible. A spanning tree in each of the two components will use up $8-2=6$ *edges. Each additional edge will add at least one cycle. Therefore a graph with 8 vertices, 2 components and 9 edges will have at least 4 cycles. 3.1.7. Impossible. A spanning tree in each of the two components will use up* $9-2=7$ *edges. Each additional edge will add at least one cycle. Therefore a graph with 9 vertices, 2 components and 10 edges will have at least 3 cycles. 3.1.8 Possible, for example, take a path with 9 vertices, add edges between* v_1 *and* v_3*, and*

v_1 and v_4. This creates 3 cycles: $v_1v_2v_3v_1$, $v_1v_2v_3v_4v_1$ and $v_1v_3v_4v_1$. 2 points.

6. Do problem 1.4.36 of the text.

   Form a bipartite graph by creating a vertex for each row and each column, and adding an edge between row i and column j if there is a brace at the intersection of the i-th row and j-th column. In this case, this graph is connected, which implies that the structure is rigid. 1 point.