Math 4707 Midterm 2

You may use books, notes, and calculators on this exam. Calculators will not be necessary. Please refrain from using other electronic devices such as laptops or cell phones. Do your work individually without collaboration. You have the full class period for the exam, and may leave early after turning in your work.

Assume that all graphs are simple.

There are 5 problems, each worth 7 points. Please write your solutions carefully, showing all your work and justifying your steps rigorously. Cite theorems and results that you use.

Problem 1. Once upon a time, there was a village with 23 gnomes. Every gnome gave hats to 5 other gnomes. Is it possible that every gnome received hats from the same 5 gnomes to whom he gave hats?
[Hint: Here and elsewhere, prove your answers!]

Solution. No. Suppose, towards a contradiction, that "giving hats" is mutual. Then we may construct a graph G whose vertex set is the 23 gnomes and an edge is present between two gnomes if and only if they presented hats to each other as presents. Being 5-regular, the number of edges is evidently $\frac{1}{2} \cdot 23 \cdot 5$, which is absurd. □

Problem 2. Suppose G is a graph and $\overline{G}$ is its complement. Prove or disprove the following statements.

(a) If G is connected, then $\overline{G}$ is connected.
(b) If G is connected, then $\overline{G}$ is disconnected.
(c) If G is disconnected, then $\overline{G}$ is connected.
(d) If G is disconnected, then $\overline{G}$ is disconnected.

Solution. Note that cycles of length 3 and 5 are counter-examples to (a) and (b), respectively. The empty graph on 23 vertices is a counter-example to (d).

Only (c) is true. Indeed, suppose G is disconnected. Let $x, y \in V(G)$ be two distinct vertices. If x and y are in the same connected component of G, pick $z \in V(G)$ in a different connected component, and note that xzy is a path in $\overline{G}$. Otherwise, x and y are not joined, so they are joined by an edge in $\overline{G}$. This means $\overline{G}$ is connected. □

Problem 3. Recall that there are n^{n-2} (labelled) trees on $[n] = \{1, 2, 3, \dots, n\}$.

(a) Determine the number of trees on $[n]$ where vertex i is a leaf for every odd integer i.

(b) Determine the number of trees on $[n]$ where the degrees of the vertices 1 and n sum up to n.

Try to write *closed-form* answers, *i.e.*, without the use of summations or variables other than n.

Solution. (a): Let $k = \lceil n/2 \rceil$ be the number of odd integers between 1 and n, inclusive. Given such a tree, remove the odd vertices to get a tree on $n - k$ vertices. There are $(n-k)^{n-k-2}$ of these. Now add the k vertices back. Each one has $n - k$ potential places to go. Therefore there are $(n-k)^{n-k-2}(n-k)^k = (n-k)^{n-2} = (n - \lceil n/2 \rceil)^{n-2} = (\lfloor n/2 \rfloor)^{n-2}$ possible trees.

(b): The numbers of edges incident to 1 and to n sum up to n. Since there are only $n - 1$ edges, (at least) one edge must be counted twice. Therefore 1 and n are joined, and all other vertices are joined to precisely one of the two. Hence there are 2^{n-2} such trees. □

Problem 4. Consider the theorem below, which is missing two important ingredients (a and b).

Theorem. *Let G be a graph on 100 vertices and 30 edges. If G has c connected components, then*

$$a \le c \le b. \qquad (*)$$

Moreover, the bounds in (*) are best possible.

First complete the statement of the theorem above by choosing appropriate integers a and b; then prove the theorem with your chosen numbers:

(a) Pick a number a for the lower bound of the number c of components in (*). Prove the assertion that $a \le c$, and then give an example of G where $c = a$ to prove the "best possible" assertion.

(b) Similarly, pick a number b for the upper bound of c in (*). Prove the assertion that $c \le b$ and give an example where $c = b$.

[Hint: Part (b) might be substantially harder than part (a).]

Solution. (a): $a = 70$. Number the connected components $1, 2, \dots, c$. Let n_i and e_i denote the number of vertices and edges in the ith component, respectively. A connected graph on n_i vertices contains a spanning tree with $n_i - 1$ edges, so $e_i \ge n_i - 1$. Summing these inequalities give

$$30 = \sum_{i=1}^{c} e_i \ge \sum_{i=1}^{c} (n_i - 1) = 100 - c,$$

so $c \ge 70$. The bound is achieved by any acyclic graph, *e.g.*, a path of length 30 with 69 isolated vertices.

(b): $b = 92$. Suppose there are $c > 92$ components. Then we may pick c vertices in different components, and add $100 - c$ vertices to existing components, one at a time. When adding a vertex to a component of size t, we can add up to t edges. Therefore this graph has at most $1 + 2 + 3 + \cdots + (100 - c)$ edges (less if we do not add vertices to the same component). Since $100 - c < 8$, we added at most $1 + 2 + \cdots + 7 = 28$ edges, a contradiction. The bound is achieved by joining a vertex to two vertices of a K_8 and adding 91 isolated vertices. □

Problem 5. Let $G = (V, E)$ be a graph and $w : E \to \mathbb{R}_+$ a weight function on its edges. For a matching $M \subseteq E$, its *weight* $w(M)$ is given by

$$w(M) = \sum_{e \in M} w(e),$$

i.e., the sum of the weights of the edges in the matching. Let $m(G)$ denote the *maximum* possible value of $w(M)$ when M ranges over all matchings of G.

Consider the following *greedy* algorithm:

(1) Start with an empty matching $M = \varnothing$.
(2) Add to M an edge e of maximum weight from $E \setminus M$ (edges not yet selected) such that M (with e added) is a matching.
(3) Repeat step (2) until such an edge e cannot be found; output the matching M.

Answer the following questions:

(a) (**5 points**) Prove that the greedy algorithm always finds a matching M such that $w(M) \geq \frac{1}{2} m(G)$.
(b) (**2 points**) Suppose that G is a bipartite graph that admits a perfect matching. Let $w(e) = 1$ for every edge $e \in E$ and run the greedy algorithm. Show that the greedy algorithm outputs a matching that covers at least half of the vertices.

[Hint: You can get **2 points** for proving (b) by using (a), even if you do not prove (a). On the other hand, if you skip (a) and prove (b) *without using* (a), you can get up to **4 points**.]

Solution. (a): Let M be an optimal matching, with $w(M) = m(G)$. Suppose $\{e_1, e_2, \ldots, e_k\}$ is an output of the greedy algorithm, where e_i is the edge chosen in round i. Consider an edge $f \in M$. If no edge e_i intersects f, then f can be added as e_{k+1} to form a better (greedy) matching, a contradiction. Thus we may let i_f be minimal such that e_{i_f} intersects f. Note that at round i_f, f could have been chosen as e_{i_f}. Therefore we conclude that $w(f) \leq w(e_{i_f})$. Summing these inequalities over all edges $f \in M$ yields

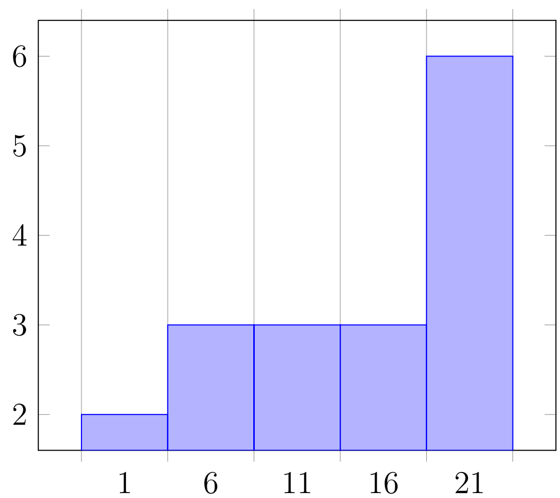
$$m(G) = w(M) = \sum_{f \in M} w(f) \leq \sum_{f \in M} w(e_{i_f}) \leq 2 \sum_{i=1}^{k} w(e_i),$$

where the last inequality is because each edge e_i can occur as e_{i_f} at most twice (once for each of its ends). This means the greedy matching $\{e_1, \ldots, e_k\}$ is at least half as good as the optimal one, as desired.

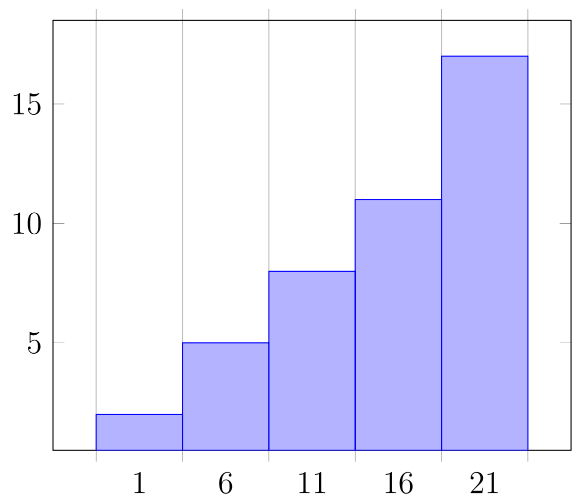
(b) from (a): If $w(e) = 1$ for every edge $e \in E$, then the weight is simply the number of edges. Suppose G is a bipartite graph with n vertices on each side. A perfect matching consists of n edges of weight 1, so $m(G) = n$. By part (a), the greedy algorithm finds a matching M with weight $w(M) \geq \frac{1}{2} m(G) = n/2$. This means M has at least $n/2$ edges (each of weight 1), and thus at least n vertices are matched, as desired.

(b) without (a): Let P be a perfect matching and M a greedy matching. Each edge $e \in P$ of the perfect matching must have (at least) one end matched by M, lest $M \cup \{e\}$ be a bigger matching, a contradiction to greediness. In other words, for each pair of vertices (pairing supplied by P), M matches at least one. This means M matches at least half of the vertices, as desired. □

Problem	Mean	Stdev
Problem 1 (7 points)	5.35	2.18
Problem 2 (7 points)	4.59	2.03
Problem 3 (7 points)	2.24	2.70
Problem 4 (7 points)	2.29	2.08
Problem 5 (7 points)	1.24	1.86
$\sum$ (35 points total)	15.71	8.02



(A) Histogram of scores.



(B) Cumulative histogram of scores.