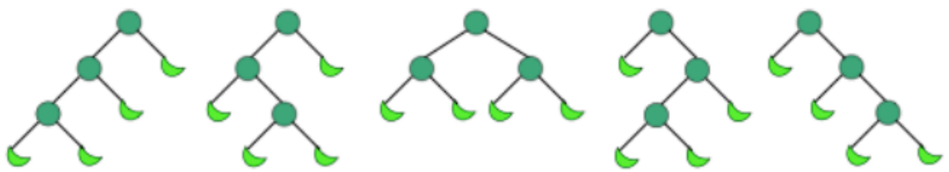
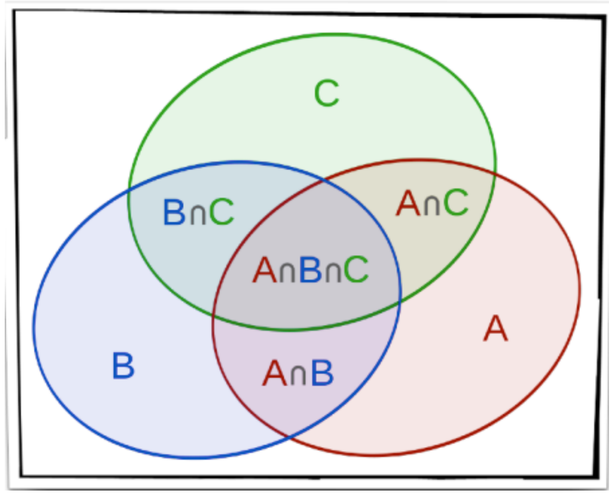
Exercise Class

Combinatorics (Spring 2024)

Teaching Assistants: 刘弘洋(liuhongyang@smail.nju.edu.cn), 王淳扬(wcysai@smail.nju.edu.cn)

balls per bin:	unrestricted	≤ 1	≥ 1
n distinct balls, m distinct bins	m^n	$(m)_n$	$m!\left\{ {n \atop m} \right\}$
n identical balls, m distinct bins	$\left(\!\binom{m}{n}\!\right)$	$\binom{m}{n}$	$\binom{n-1}{m-1}$
n distinct balls, m identical bins	$\sum_{k=1}^{m} \left\{ {n \atop k} \right\}$	$\begin{cases} 1 & \text{if } n \leq m \\ 0 & \text{if } n > m \end{cases}$	$\left\{ {n \atop m} \right\}$
n identical balls, m identical bins	$\sum_{k=1}^{m} p_k(n)$	$\begin{cases} 1 & \text{if } n \leq m \\ 0 & \text{if } n > m \end{cases}$	$p_m(n)$





Problem Set 1

basic enumeration, generating functions, sieve methods

Problem 1

How many $n \times m$ matrices of 0's and 1's are there, such that every row and column contains an even number of 1's? An odd number of 1's?

1. Note that after arbitrarily filling the upper left $(n-1) \times (m-1)$ submatrix, there's exactly one way to fill the remaining such that all constraints are satisfied, hence there are $2^{(n-1)\times(m-1)}$ ways.

2. When the parity of n and m differs, there clearly don't exist any solution. Otherwise there are $2^{(n-1)\times(m-1)}$ following the same argument as the previous question.

Problem 2

- There is a set of $2n$ people: n male and n female. A good party is a set with the same number of male and female. How many possibilities are there to build such a good party?
- Try to express the answer using one binomial coefficient and provide a combinatorial proof.

1. Consider enumerating on the number of male/females, we obtain $\sum_{i=0}^{n} \binom{n}{i}^2$.

2. We have $\sum_{i=0}^{n} \binom{n}{i}^2 = \binom{2n}{n}$ (Vandermonde's identity). A combinatorial proof would be constructing a bijection by considering choosing the set of males in the party and the set of females not in the party, which would be exactly n people out of $2n$.

Problem 3

Given $0 \leq k \leq n$, prove that

$$\sum_{i=0}^{\lfloor k/2 \rfloor} \binom{n}{k-i} \binom{k-i}{i} = \sum_{j=0}^{\lfloor k/3 \rfloor} (-1)^j \binom{n}{j} \binom{k-3j+n-1}{n-1}.$$

(Hint: Consider the number of ways to to place k indistinguishable balls into n boxes, with no more than 2 balls in each box.)

Let the number of ways to place k indistinguishable balls into n boxes, with no more than 2 balls in each box be S. We show that both LHS and RHS equal to S.

LHS: enumerate over i, the number of boxes with 2 balls, the expression then first considers choosing the set of boxes with at least one ball (there are $k - i$ such boxes), then considers choosing the set of boxes with 2 balls out of those $k - i$ boxes.

RHS: For each box i we let A_i be the property that box i has more than 2 balls. Applying the Principle of Inclusion-Exclusion then suffices. (enumerate over j, the number of violated properties, there are $\binom{n}{j}$ of those, and simultaneously removing 3 balls from these boxes then applying stars-and-bars we arrive at the RHS expression)

Problem 4

- Suppose $n, k \geq 1$ and $j = \lfloor k/2 \rfloor$. Let $S(n,k)$ denote a Stirling number of the second kind. Give a generating function proof that

$$S(n,k) \equiv \binom{n-j-1}{n-k} \pmod{2}.$$

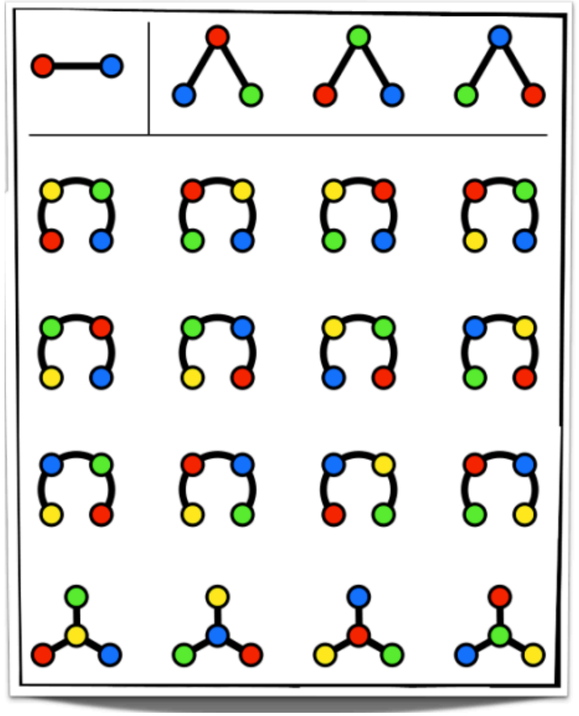
- State and prove an analogous result for Stirling numbers of the first kind.

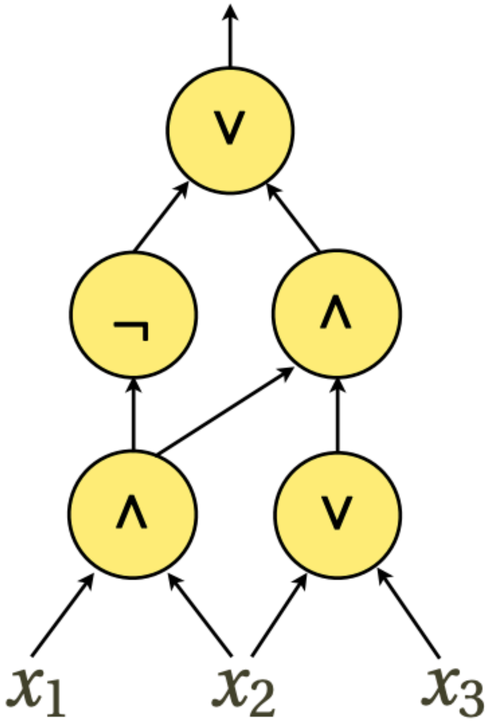
Problem 5

For the following problems, provide a formula and explain your answer.

- Count the number of ways to place n chess pieces on an $n \times n$ chessboard such that each row, each column, and **the main diagonal** have at least one chess piece.
- Let n be even. Count the number of ways to place n chess pieces on an $n \times n$ chessboard such that each row, each column, and **both diagonals** have at least one chess piece.

1. The number is equal to the number of permutations π that has at least one positions i with $\pi(i) = i$, which equals to $n! - D_n$, where D_n is the number of derangements of a set of size n.



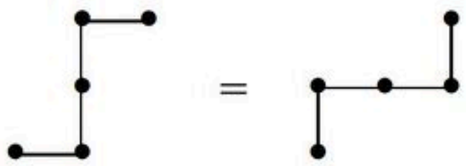


Problem Set 2

Polya's theory of counting, Cayley's formula, existence by counting

Problem 1

Count the number of lattice paths from $(0,0)$ to (n,n) with steps up and to the right, when two paths are considered equal if one can be moved on top of the other by a rotation or reflection. For example:



(Hint: Represent a path as a sequence of 0s and 1s and determine the group that acts on these sequences.)

The lattice paths correspond to sequences of n 0's and n 1's, where 0 represents an (1,0)-step and 1 a (0,1)-step. The group actions consist of

- the identity;
- the reverse $\sigma : a_1 \dots a_{2n} \to a_{2n} \dots a_1$;
- the exchange $\varepsilon : a_1 \dots a_{2n} \to \overline{a_1} \dots \overline{a_{2n}}$, where $\overline{a_i} \neq a_i$;
- and $\sigma\varepsilon$.

Note that it's easy to calculate that:

$$|X_{\text{id}}| = \binom{2n}{n}, |X_\sigma| = \binom{n}{n/2} \cdot [n \text{ even}], |X_\varepsilon| = 0, |X_{\sigma\varepsilon}| = 2^n,$$

Hence by Burnside's Lemma, we have the answer equals to

$$|X \backslash G| = \frac{1}{|G|} \sum_{\pi \in G} |X_\pi| = \frac{1}{4} \left(\binom{2n}{n} + 2^n + \binom{n}{n/2} \cdot [n \text{ even}] \right).$$

Problem 2

Suppose we are given m n-sets whose elements are colored with two colors. Two colorings are equal if they arise from each other by a permutation of the m sets followed by permutations of the individual sets. Determine the number of different colorings using Pólya's enumeration formula. (The answer should be easily verifiable through combinatorial arguments if you get it right!)

The permutation group G is $S(m)[S(n)]$. Hence, the desired number is

$$|X\backslash G| = \frac{1}{|G|}\sum_{\pi\in G}|X_\pi| = \frac{1}{m!(n!)^m}\sum_{\pi^*\in S_m}\sum_{\pi_1,\pi_2,\ldots,\pi_m\in S_n}|X_{\pi^*,\pi_1,\pi_2,\ldots,\pi_m}| = \frac{1}{m!(n!)^m}\sum_{\pi^*\in S_m}\prod_{c\in C(\pi^*)}\sum_{\pi_1,\pi_2,\ldots,\pi_{|c|}\in S_n}2^{C(\pi_1,\pi_2,\ldots,\pi_{|c|})}.$$

Here it's easy to note that for any π^* and $c \in C(\pi^*)$,

$$\sum_{\pi_1,\pi_2,\ldots,\pi_{|c|}\in S_n}2^{C(\pi_1,\pi_2,\ldots,\pi_{|c|})} = (n!)^{|c|-1}\sum_{\pi\in S_n}2^{|C(\pi)|} = (n!)^{|c|}\cdot(n+1),$$

where the last equality is by applying Burnside's lemma on the number of colorings on one n-set.

Therefore, we have

$$|X\backslash G| = \frac{1}{m!}\sum_{\pi^*\in S_m}(n+1)^{|C(\pi)|} = \frac{1}{m!}\sum_{k=1}^{m}s_{m,k}(n+1)^k = \frac{1}{m!}(n+1)^{\overline{m}} = \binom{n+m}{n}.$$

Problem 3

Prove the following claims related to Cayley's formula:

- The number of rooted labelled forests with n vertices is given by $(n+1)^{n-1}$.
- The number of unrooted labelled forests with n vertices and k connected components such that $1, \ldots, k$ belong to distinct components is given by kn^{n-k-1}.
- The number of unrooted labelled trees with n vertices of degrees $d_1, d_2, \ldots, d_n$ respectively is given by $\binom{n-2}{d_1-1, d_2-1, \ldots, d_n-1} = \frac{(n-2)!}{(d_1-1)!(d_2-1)!\cdots(d_n-1)!}$.

1. Consider adding a new “superroot” vertex and connect all roots of the forest to this “super root”.

It’s easy to show that this creates a bijection between the number of rooted forests with n vertices and the number of unrooted trees with $n + 1$ follows, then the claim directly follows from Cayley’s formula.

Problem 3

Prove the following claims related to Cayley's formula:

- The number of rooted labelled forests with n vertices is given by $(n+1)^{n-1}$.
- The number of unrooted labelled forests with n vertices and k connected components such that $1, \ldots, k$ belong to distinct components is given by kn^{n-k-1}.
- The number of unrooted labelled trees with n vertices of degrees $d_1, d_2, \ldots, d_n$ respectively is given by $\binom{n-2}{d_1-1, d_2-1, \ldots, d_n-1} = \frac{(n-2)!}{(d_1-1)!(d_2-1)!\cdots(d_n-1)!}$.

2. We let the desired number be $T_{n,k}$ and double count the number of unrooted labelled trees with $n+1$ vertices, such that the degree of vertex $n+1$ is k:

(1) Choose any k distinct vertices $v_1, v_2, \ldots, v_k$ from $[n]$. Build a unrooted labelled forest with n vertices and k connected components such that $v_1, v_2, \ldots, v_k$ belong to different components, connect to vertex $n+1$. This gives $T_{n,k} \cdot (n+1)$ number of ways.

(2) Consider directly building a labelled tree with the degree restriction on vertex $n+1$. Note that this means that $n+1$ must appear $k-1$ times in the Prüfer code. This gives $\binom{n-1}{k-1} \cdot n^{n-k}$ number of ways.

(Other possible methods include modifying Prüfer code, induction)

Problem 3

Prove the following claims related to Cayley's formula:

- The number of rooted labelled forests with n vertices is given by $(n+1)^{n-1}$.
- The number of unrooted labelled forests with n vertices and k connected components such that $1, \ldots, k$ belong to distinct components is given by kn^{n-k-1}.
- The number of unrooted labelled trees with n vertices of degrees $d_1, d_2, \ldots, d_n$ respectively is given by $\binom{n-2}{d_1-1, d_2-1, \ldots, d_n-1} = \frac{(n-2)!}{(d_1-1)!(d_2-1)!\cdots(d_n-1)!}$.

3. Consider the Prüfer code of the tree. The degree restrictions can be converted to:

- For each $1 \leq i \leq n$, i appears exactly $d_i - 1$ times in the Prüfer code.

Then the claim directly follows.

Problem 4

There are n parking spaces $1, 2, \ldots, n$ available for n drivers. Each driver has a favorite space, driver i space $f(i)$, $1 \leq f(i) \leq n$. The drivers arrive one by one. When driver i arrives he tries to park his car in space $f(i)$. If it is taken he moves down the line to take the first free space greater than $f(i)$, if any. Example: $n = 4, f = 3221$; then driver $1 \to 3, 2 \to 2, 3 \to 4, 4 \to 1$, but for $f = 2332$ we have $1 \to 2, 2 \to 3, 3 \to 4, 2 \to ?$. Let $p(n)$ be the number of sequences f that allow each driver to park his car; f is then called a parking sequence. Prove:

- f is a parking sequence if and only if $\#\{i : f(i) \leq k\} \geq k$.
- $p(n) = (n+1)^{n-1}$. This looks like the number of rooted labelled forests with n vertices. Can you find a bijection?

1. Both directions are easy following the definition. We simply skip it.

2. Add an additional space $n+1$ and make parking a cycle. Now all cars can park, and always one space remains. Notice, that f is a parking function if and only if the empty space is $n+1$. Then by symmetry we have $p(n) = \dfrac{(n+1)^n}{(n+1)} = (n+1)^{n-1}$. Note that this proof also show that for any f, exactly one of $(f_1 + p, f_2 + p, \ldots, f_n + p) \pmod{n+1}$ is a parking function.

   For constructing bijection, let f be a parking function and consider the sequence $d \in [n+1]^{n-1}$

   such that for each $i \in [n-1]$, $d_i = f_{i+1} - f_i \pmod{n+1}$. By the argument above, this gives a bijection between all parking functions and all sequences in $[n+1]^{n-1}$, then a bijection can be easily constructed using the Prüfer code.

Problem 5

Let's dive into the world of (one-way) communication complexity. Alice has an input $\mathbf{x} \in \{0,1\}^a$, Bob has an input $\mathbf{y} \in \{0,1\}^b$, and the goal is to compute a Boolean function $f : \{0,1\}^a \times \{0,1\}^b \to \{0,1\}$ of the joint input $(\mathbf{x}, \mathbf{y})$. The players communicate as follows: Alice sends a message $\mathbf{z}$ to Bob as a function of $\mathbf{x}$ only (she doesn't know Bob's input $\mathbf{y}$), and Bob has to decide the function f knowing only $\mathbf{z}$ and $\mathbf{y}$ (he doesn't know Alice's input x). The one-way communication complexity of f is the smallest number of bits communicated (in the worst case over $(\mathbf{x}, \mathbf{y})$) of any protocol that computes f.

We consider the (one-way) communication complexity of a problem called INDEX. In an instance of INDEX, Alice gets an n-bit string $\mathbf{x} \in \{0,1\}^n$ and Bob gets an integer $i \in \{1, 2, \ldots, n\}$, encoded in binary using $\approx \log_2 n$ bits. The goal is simply to compute x_i, the ith bit of Alice's input.

- Show that the deterministic (one-way) communication complexity of INDEX is $\Omega(n)$. (For a deterministic communication protocol, the protocol is considered to solve INDEX if Bob always outputs the correct answer in the worst case over $(\mathbf{x}, \mathbf{y})$);
- Show that the randomized (one-way) communication complexity of INDEX is $\Omega(n)$. (In a randomized communication protocol, Alice and Bob have access to *public random bits*: a deity writes an infinite sequence of perfectly random bits on a blackboard visible to both Alice and Bob. Alice and Bob can freely use as many of these random bits as they want — it doesn't contribute to the communication cost of the protocol. The protocol is considered to solve INDEX if Bob outputs the correct answer with probability at least $2/3$ in the worst case over $(\mathbf{x}, \mathbf{y})$).

(Hint: We need to formalize the intuition that Bob typically (over $\mathbf{x}$) doesn't learn very much about $\mathbf{x}$, and hence typically (over i) doesn't know what x_i is. Try some counting arguments.)

1. Assume (for the sake of contradiction) the one-way deterministic communication complexity is at most $n-1$. Then Alice sends at most $n-1$ bits and 2^{n-1} distinct messages. Since Alice has 2^n possible inputs, then by the Pigeonhole Principle, there must exist two distinct inputs $\mathbf{x}_0$ and $\mathbf{x}_1$ where Alice sends the same message. Let i denote an index where $\mathbf{x}_0$ and $\mathbf{x}_1$ differ. When Bob receives the index i, he will be wrong for exactly one of the cases $\mathbf{x} = \mathbf{x}_0$ or $\mathbf{x} = \mathbf{x}_1$. Hence we conclude such protocol cannot exist.

Problem 5

Let's dive into the world of (one-way) communication complexity. Alice has an input $\mathbf{x} \in \{0,1\}^a$, Bob has an input $\mathbf{y} \in \{0,1\}^b$, and the goal is to compute a Boolean function $f : \{0,1\}^a \times \{0,1\}^b \to \{0,1\}$ of the joint input $(\mathbf{x}, \mathbf{y})$. The players communicate as follows: Alice sends a message $\mathbf{z}$ to Bob as a function of $\mathbf{x}$ only (she doesn't know Bob's input $\mathbf{y}$), and Bob has to decide the function f knowing only $\mathbf{z}$ and $\mathbf{y}$ (he doesn't know Alice's input x). The one-way communication complexity of f is the smallest number of bits communicated (in the worst case over $(\mathbf{x}, \mathbf{y})$) of any protocol that computes f.

We consider the (one-way) communication complexity of a problem called INDEX. In an instance of INDEX, Alice gets an n-bit string $\mathbf{x} \in \{0,1\}^n$ and Bob gets an integer $i \in \{1, 2, \ldots, n\}$, encoded in binary using $\approx \log_2 n$ bits. The goal is simply to compute x_i, the ith bit of Alice's input.

- Show that the deterministic (one-way) communication complexity of INDEX is $\Omega(n)$. (For a deterministic communication protocol, the protocol is considered to solve INDEX if Bob always outputs the correct answer in the worst case over $(\mathbf{x}, \mathbf{y})$);
- Show that the randomized (one-way) communication complexity of INDEX is $\Omega(n)$. (In a randomized communication protocol, Alice and Bob have access to *public random bits*: a deity writes an infinite sequence of perfectly random bits on a blackboard visible to both Alice and Bob. Alice and Bob can freely use as many of these random bits as they want — it doesn't contribute to the communication cost of the protocol. The protocol is considered to solve INDEX if Bob outputs the correct answer with probability at least $2/3$ in the worst case over $(\mathbf{x}, \mathbf{y})$).

(Hint: We need to formalize the intuition that Bob typically (over $\mathbf{x}$) doesn't learn very much about $\mathbf{x}$, and hence typically (over i) doesn't know what x_i is. Try some counting arguments.)

2. We let the distribution of the input $\mathbf{x}$ Alice receives be chosen uniformly from $\{0,1\}^n$ and the distribution of the input $\mathbf{y}$ Bob receives be chosen uniformly from $[n]$. We'll show that for sufficiently large n, every deterministic one-way protocol P that uses at most $0.1n$ bits of communication has error at least $1/8$. This suffices to prove the randomized one-way communication complexity of INDEX because

   (1) under this setting a randomized protocol can be considered as a weighted average of determinstic protocols

   (2) all constant error probabilities $\varepsilon \in (0, \frac{1}{2})$ yield the same communication complexity, up to a constant factor, as the success probability of a protocol can be boosted through amplification (i.e., repeated trials). (If you don't get it, consider taking Advanced Algorithms course in the upcoming 2024 Fall!)

For simplicity we'll omit the detailed proof for these two arguments here.

Problem 5

Let's dive into the world of (one-way) communication complexity. Alice has an input $\mathbf{x} \in \{0,1\}^a$, Bob has an input $\mathbf{y} \in \{0,1\}^b$, and the goal is to compute a Boolean function $f : \{0,1\}^a \times \{0,1\}^b \to \{0,1\}$ of the joint input $(\mathbf{x}, \mathbf{y})$. The players communicate as follows: Alice sends a message $\mathbf{z}$ to Bob as a function of $\mathbf{x}$ only (she doesn't know Bob's input $\mathbf{y}$), and Bob has to decide the function f knowing only $\mathbf{z}$ and $\mathbf{y}$ (he doesn't know Alice's input x). The one-way communication complexity of f is the smallest number of bits communicated (in the worst case over $(\mathbf{x}, \mathbf{y})$) of any protocol that computes f.

We consider the (one-way) communication complexity of a problem called INDEX. In an instance of INDEX, Alice gets an n-bit string $\mathbf{x} \in \{0,1\}^n$ and Bob gets an integer $i \in \{1, 2, \ldots, n\}$, encoded in binary using $\approx \log_2 n$ bits. The goal is simply to compute x_i, the ith bit of Alice's input.

- Show that the deterministic (one-way) communication complexity of INDEX is $\Omega(n)$. (For a deterministic communication protocol, the protocol is considered to solve INDEX if Bob always outputs the correct answer in the worst case over $(\mathbf{x}, \mathbf{y})$);
- Show that the randomized (one-way) communication complexity of INDEX is $\Omega(n)$. (In a randomized communication protocol, Alice and Bob have access to *public random bits*: a deity writes an infinite sequence of perfectly random bits on a blackboard visible to both Alice and Bob. Alice and Bob can freely use as many of these random bits as they want — it doesn't contribute to the communication cost of the protocol. The protocol is considered to solve INDEX if Bob outputs the correct answer with probability at least $2/3$ in the worst case over $(\mathbf{x}, \mathbf{y})$).

(Hint: We need to formalize the intuition that Bob typically (over $\mathbf{x}$) doesn't learn very much about $\mathbf{x}$, and hence typically (over i) doesn't know what x_i is. Try some counting arguments.)

2. (cont'd) Suppose Bob gets a message $\mathbf{z}$ from Alice. Since the protocol is deterministic, Bob has to announce a bit, "0" or "1," as a function of $\mathbf{z}$ and i only. Fix $\mathbf{z}$ and let $\mathbf{a}(\mathbf{z})$ be an n-bit vector such that $\mathbf{a}(\mathbf{z})_i$ is Bob's output when he receives message $\mathbf{z}$ and input $\mathbf{y} = i$. The protocol is correct if Bob holds an input i with $\mathbf{a}(\mathbf{z})_i = \mathbf{x}_i$, and incorrect otherwise. Note that Bob's index i is chosen uniformly at random, and independently of $\mathbf{x}$, we have

$$\Pr_i[P \text{ is incorrect} \mid \mathbf{x}, \mathbf{z}] = \frac{d_H(\mathbf{a}, \mathbf{z})}{n},$$

where $d_H(\mathbf{x}, \mathbf{a}(\mathbf{z}))$ denotes the Hamming distance between the vectors $\mathbf{x}$ and $\mathbf{a}(\mathbf{z})$ (i.e., the number of coordinates they differ). Our goal is to show that, with constant probability over the choice of $\mathbf{x}$, the above expression is bounded below by a constant, which together gives a 1/8 lower bound for error probability.

Problem 5

Let's dive into the world of (one-way) communication complexity. Alice has an input $\mathbf{x} \in \{0,1\}^a$, Bob has an input $\mathbf{y} \in \{0,1\}^b$, and the goal is to compute a Boolean function $f : \{0,1\}^a \times \{0,1\}^b \to \{0,1\}$ of the joint input $(\mathbf{x}, \mathbf{y})$. The players communicate as follows: Alice sends a message $\mathbf{z}$ to Bob as a function of $\mathbf{x}$ only (she doesn't know Bob's input $\mathbf{y}$), and Bob has to decide the function f knowing only $\mathbf{z}$ and $\mathbf{y}$ (he doesn't know Alice's input x). The one-way communication complexity of f is the smallest number of bits communicated (in the worst case over $(\mathbf{x}, \mathbf{y})$) of any protocol that computes f.

We consider the (one-way) communication complexity of a problem called INDEX. In an instance of INDEX, Alice gets an n-bit string $\mathbf{x} \in \{0,1\}^n$ and Bob gets an integer $i \in \{1, 2, \ldots, n\}$, encoded in binary using $\approx \log_2 n$ bits. The goal is simply to compute x_i, the ith bit of Alice's input.

- Show that the deterministic (one-way) communication complexity of INDEX is $\Omega(n)$. (For a deterministic communication protocol, the protocol is considered to solve INDEX if Bob always outputs the correct answer in the worst case over $(\mathbf{x}, \mathbf{y})$);
- Show that the randomized (one-way) communication complexity of INDEX is $\Omega(n)$. (In a randomized communication protocol, Alice and Bob have access to *public random bits*: a deity writes an infinite sequence of perfectly random bits on a blackboard visible to both Alice and Bob. Alice and Bob can freely use as many of these random bits as they want — it doesn't contribute to the communication cost of the protocol. The protocol is considered to solve INDEX if Bob outputs the correct answer with probability at least $2/3$ in the worst case over $(\mathbf{x}, \mathbf{y})$).

(Hint: We need to formalize the intuition that Bob typically (over $\mathbf{x}$) doesn't learn very much about $\mathbf{x}$, and hence typically (over i) doesn't know what x_i is. Try some counting arguments.)

2. (cont'd) Since there are at most $2^{0.1n}$ possible messages $\mathbf{z}$, there are at most $2^{0.1n}$ possible answer vectors $\mathbf{a}(\mathbf{z})$. Let $A = \{\mathbf{a}(\mathbf{z}(\mathbf{x})) : \mathbf{x} \in \{0,1\}^n\}$ denote the set of all answer vectors used by the protocol P. Then we have $|A| \leq 2^{0.1n}$. Call Alice's input $\mathbf{x}$ good if there exists an answer vector $\mathbf{a}$ with $d_H(\mathbf{x}, \mathbf{a}) < \frac{n}{4}$, and bad otherwise. We claim that for sufficiently large n, there are at least 2^{n-1} bad inputs. This claim already implies what we want to prove since now the error probability is at least $\frac{1}{2} \times \frac{1}{4} = \frac{1}{8}$.

   Intuition of the claim: Think of each answer vector $\mathbf{a}$ as the center of a ball of radius $n/4$ in the Hamming cube. Because there aren't too many balls (only $2^{0.1n}$) and their radii aren't too big (only $n/4$), the union of all of the balls is less than half of the Hamming cube.

Problem 5

Let's dive into the world of (one-way) communication complexity. Alice has an input $\mathbf{x} \in \{0,1\}^a$, Bob has an input $\mathbf{y} \in \{0,1\}^b$, and the goal is to compute a Boolean function $f : \{0,1\}^a \times \{0,1\}^b \to \{0,1\}$ of the joint input $(\mathbf{x}, \mathbf{y})$. The players communicate as follows: Alice sends a message $\mathbf{z}$ to Bob as a function of $\mathbf{x}$ only (she doesn't know Bob's input $\mathbf{y}$), and Bob has to decide the function f knowing only $\mathbf{z}$ and $\mathbf{y}$ (he doesn't know Alice's input x). The one-way communication complexity of f is the smallest number of bits communicated (in the worst case over $(\mathbf{x}, \mathbf{y})$) of any protocol that computes f.

We consider the (one-way) communication complexity of a problem called INDEX. In an instance of INDEX, Alice gets an n-bit string $\mathbf{x} \in \{0,1\}^n$ and Bob gets an integer $i \in \{1, 2, \ldots, n\}$, encoded in binary using $\approx \log_2 n$ bits. The goal is simply to compute x_i, the ith bit of Alice's input.

- Show that the deterministic (one-way) communication complexity of INDEX is $\Omega(n)$. (For a deterministic communication protocol, the protocol is considered to solve INDEX if Bob always outputs the correct answer in the worst case over $(\mathbf{x}, \mathbf{y})$);
- Show that the randomized (one-way) communication complexity of INDEX is $\Omega(n)$. (In a randomized communication protocol, Alice and Bob have access to *public random bits*: a deity writes an infinite sequence of perfectly random bits on a blackboard visible to both Alice and Bob. Alice and Bob can freely use as many of these random bits as they want — it doesn't contribute to the communication cost of the protocol. The protocol is considered to solve INDEX if Bob outputs the correct answer with probability at least $2/3$ in the worst case over $(\mathbf{x}, \mathbf{y})$).

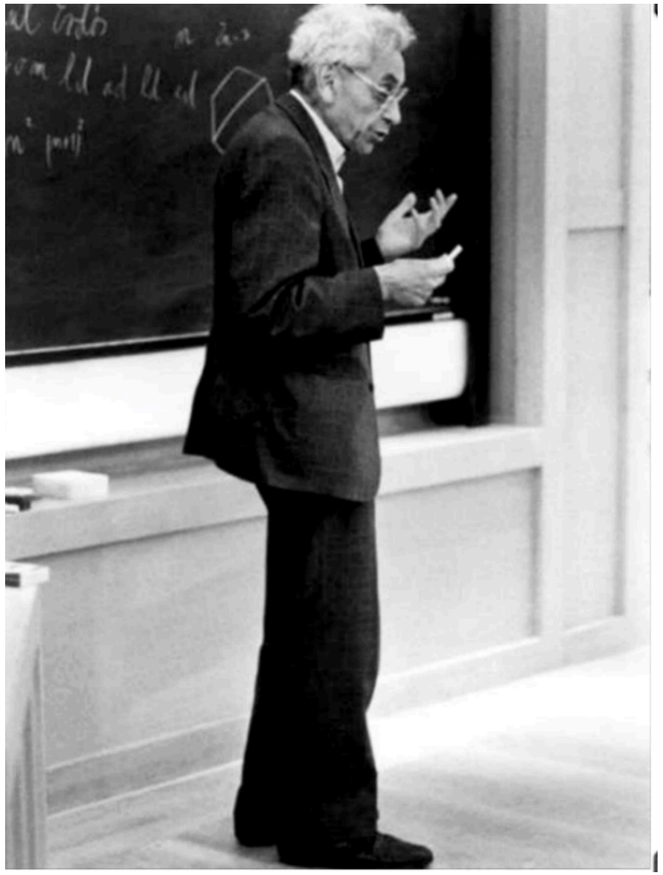
(Hint: We need to formalize the intuition that Bob typically (over $\mathbf{x}$) doesn't learn very much about $\mathbf{x}$, and hence typically (over i) doesn't know what x_i is. Try some counting arguments.)

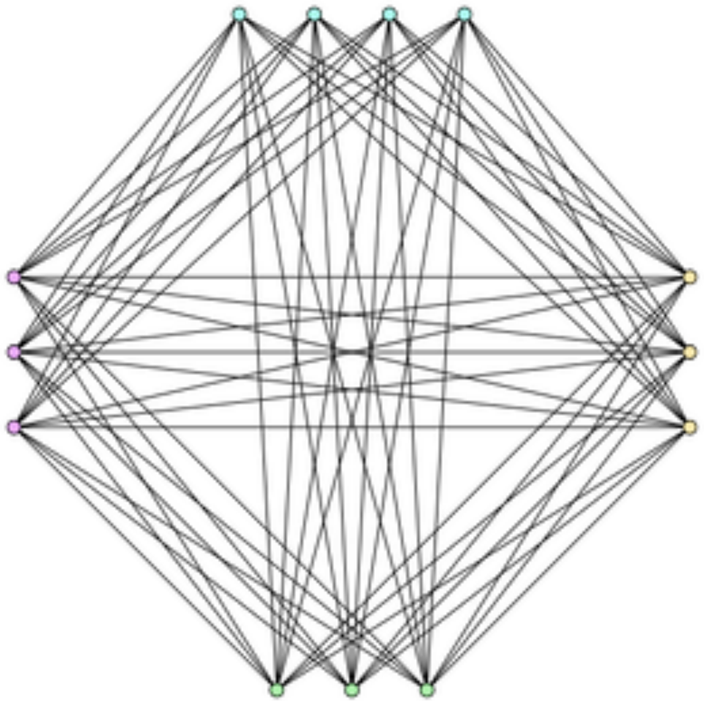
2. (cont'd) We then prove the claim using a counting argument. Fix some answer vector $\mathbf{a} \in A$. The number of inputs $\mathbf{x}$ with Hamming distance at most $n/4$ from $\mathbf{a}$ is

$$\sum_{i=0}^{n/4} \binom{n}{i} \leq n(4\mathrm{e})^{n/4} \leq n2^{0.861n},$$

Which finishes the proof. Here, the first inequality is by Stirling's approximation that

$$\binom{n}{k} \leq \left(\frac{\mathrm{e}n}{k}\right)^k,$$





Problem Set 3

existence problems, probabilistic methods, extremal graph theory

Problem 1

Solve the following two existence problems:

- You are given n integers $a_1, a_2, \ldots, a_n$, such that for each $1 \leq i \leq n$ it holds that $i - n \leq a_i \leq i - 1$. Show that there exists a **nonempty** subsequence (not necessarily consecutive) of these integers, whose sum is equal to 0. (Hint: Consider $b_i = a_i - i$)
- You are given two **multisets** A and B, both with n integers from 1 to n. Show that there exist two **nonempty** subsets $A' \subseteq A$ and $B' \subseteq B$ with equal sum, i.e. $\sum_{x \in A'} x = \sum_{y \in B'} y$ (Hint: Replace the term *multiset* by *sequence*, the term *subset* by *consecutive subsequence*, and the statement is still true.)

1. For each $1 \leq i \leq n$, let $b_i = a_i - i$, then we have $-n \leq b_i \leq -1$. Consider the directed graph $G = (V, E)$ where $V = [n]$ and $E = \{(-i, b_i) \mid 1 \leq i \leq n\}$, then the outdegree of each vertex is exactly 1. Hence, there must exist a cycle $C = (v_0, v_1, \ldots, v_\ell)$ such that $(v_i, v_{(i+1) \bmod \ell}) \in E$ for each $0 \leq i \leq \ell$. Therefore,

$$\sum_{0 \leq i \leq \ell} a_{v_i} = \sum_{0 \leq i \leq \ell} (v_i - b_{v_i}) = \sum_{0 \leq i \leq \ell} v_i - \sum_{0 \leq i \leq \ell} b_{v_i} = 0,$$

and the claim is proved.

Problem 1

Solve the following two existence problems:

- You are given n integers $a_1, a_2, \ldots, a_n$, such that for each $1 \leq i \leq n$ it holds that $i - n \leq a_i \leq i - 1$. Show that there exists a **nonempty** subsequence (not necessarily consecutive) of these integers, whose sum is equal to 0. (Hint: Consider $b_i = a_i - i$)
- You are given two **multisets** A and B, both with n integers from 1 to n. Show that there exist two **nonempty** subsets $A' \subseteq A$ and $B' \subseteq B$ with equal sum, i.e. $\sum_{x \in A'} x = \sum_{y \in B'} y$ (Hint: Replace the term *multiset* by *sequence*, the term *subset* by *consecutive subsequence*, and the statement is still true.)

2. Assume $A = \{A_1, A_2, \ldots, A_n\}, B = \{B_1, B_2, \ldots, B_n\}$. For any $0 \leq i \leq n$, let $a_i = \sum_{j=1}^{i} A_i, b_i = \sum_{j=1}^{i} B_i$.

If we can find $0 \leq i < i' \leq n$,$0 \leq j < j' \leq n$ such that $a_{i'} - a_i = b_{j'} - b_j$, then we have found a solution by letting $A' = \{A_{i'+1}, A_{i'+2}, \ldots, A_i\}, B' = \{B_{j'+1}, B_{j'+2}, \ldots, B_j\}$.

We then prove that the desired tuple (i, i', j, j') must exist. Assume without loss of generality that $a_n \leq b_n$. For each $0 \leq i \leq n$, we can find the largest $0 \leq j \leq n$ such that $b_j \leq a_i$, call such j as $f(i)$. Note that by this definition we have $0 \leq a_i - b_{f(i)} \leq n - 1$for each $0 \leq i \leq n$. Therefore by the pigeonhole principle, there must exist $0 \leq i < i' \leq n$ such that $a_i - b_{f(i)} = a_{i'} - b_{f(i')}$, then $(i, i', f(i), f(i'))$ is the desired tuple.

Problem 2

Prove that there is an absolute constant $c > 0$ with the following property: let A be an $n \times n$ matrix with pairwise distinct entries. Then there is a permutation of the rows of A so that no column in the permuted matrix contains an increasing subsequence of length at least $c\sqrt{n}$.

We permute the rows of A uniformly at random. Let $\mathscr{E}_i^k$ be the event that the i-th column contains an increasing subsequence of length k. We then bound $\mathbf{Pr}[\mathscr{E}_i^k]$. Note that for any fixed subset of numbers in the i-th column of size k, the probability of it forming an increasing subsequence is $1/k!$. Hence by union bound and Stirling's approximation we have

$$\mathbf{Pr}[\mathscr{E}_i^k] \leq \binom{n}{k} \cdot \frac{1}{k!} < \frac{n^k}{(k!)^2} < \frac{n^2}{(k/\mathrm{e})^{2k}} = \left(\frac{\mathrm{e}^2 n}{k^2}\right)^k.$$

Let $k = 2\mathrm{e}\sqrt{n}$, then we have

$$\mathbf{Pr}[\mathscr{E}_i^k] \leq 2^{-4\mathrm{e}\sqrt{n}}, \quad \mathbf{Pr}\left[\bigvee_{i=1}^{n} \mathscr{E}_i^k\right] \leq n2^{-4\mathrm{e}\sqrt{n}},$$

therefore a valid permutation always exists.

Problem 3

A 3-uniform hypergraph $H = (V, E)$ consists of a set of vertices V and a set of edges E, where each edge $e \in E$ is a set of exactly three vertices from V. Prove:

Every 3-uniform hypergraph with n vertices and $m \geq n/3$ edges contains an independent set (i.e., a set of vertices containing no edges) of size at least $\frac{2n^{1.5}}{3\sqrt{3m}}$.

We construct an independent set I as follows. First choose each vertex u.a.r with probability p, then for each $e \in E$, remove e if $e \subseteq I$. Then it is clear that I is an independent set and we have

$$\mathbb{E}[\,|I|\,] \geq np - mp^3\,.$$

Set $p = \sqrt{\frac{n}{3m}}$, we have that $\mathbb{E}[\,|I|\,] \geq \frac{2n^{1.5}}{3\sqrt{3m}}$, finishing the proof.

Problem 4

Let $D = (V, E)$ be a simple directed graph with **minimum outdegree** δ and **maximum indegree** Δ. Prove:

For any positive integer k, if $e(\Delta\delta + 1)(1 - \frac{1}{k})^{\delta} < 1$, then D contains a (directed,simple) cycle of length $0 \pmod{k}$.

Hint: Let $f : V \to \{0, 1, \ldots, k-1\}$ be a random coloring of V obtained by choosing uniformly and independently at random for each $v \in V$. Try to use Lovász Local Lemma to show something interesting!

Clearly, we may assume that every outdegree is precisely δ, since otherwise we can consider a subgraph of D with this property.

Let $f : V \to \{0,1,\ldots,k-1\}$ be a random coloring of V, obtained by choosing uniformly and independently at random for each $v \in V$. For each $v \in V$, let A_v denote the event where there is no $u \in V$, with $(v, u) \in E$ and $f(u) \equiv f(v) + 1 \pmod{k}$. Clearly, $\mathbf{Pr}[A_v] = (1 - 1/k)^{\delta}$. One can easily check that each event A_v is mutually independent of all the events A_u but those satisfying

$$N^+(v) \cap \{u \cup N^+(v)\} \neq \varnothing,$$

where $N^+(v) = \{w \in V : (v, w) \in E\}$. The number of such u's is at most $\Delta\delta$ and hence

$\mathbf{Pr}\left[\bigwedge_{v \in V} \neg A_v\right] > 0$, by the Lovász Local Lemma.

Then, there exists a coloring f such that no A_v occurs. Start from an arbitrary vertex v and repeatedly go to $u \in V$, with $(v, u) \in E$ and $f(u) \equiv f(v) + 1 \pmod{k}$. One eventually visits a vertex that have been visited, which yields a directed simple cycle of length $0 \pmod{k}$.

Problem 5

Let $G = (V, E)$ be a graph on n vertices and $t(G)$ the number of triangles in it. Show that $t(G) \geq \frac{|E|}{3n}(4|E| - n^2)$.

Hint: For an edge $e = (u, v) \in E$, let $t(e)$ be the number of triangles containing e. Try to obtain that $t(e) \geq d(u) + d(v) - n$ and use Cauchy–Schwarz inequality to esitimate the sum $\sum_{e=(u,v)\in E} (d(u) + d(v))$.

For an edge $e = (u, v) \in E$, let $X = N(u)\backslash\{v\}$ and $Y = N(v)\backslash\{u\}$. Note that (u, v, w) is a triangle for any $w \in X \cap Y$. Note that $|X \cup Y| \leq n - 2$. Hence,

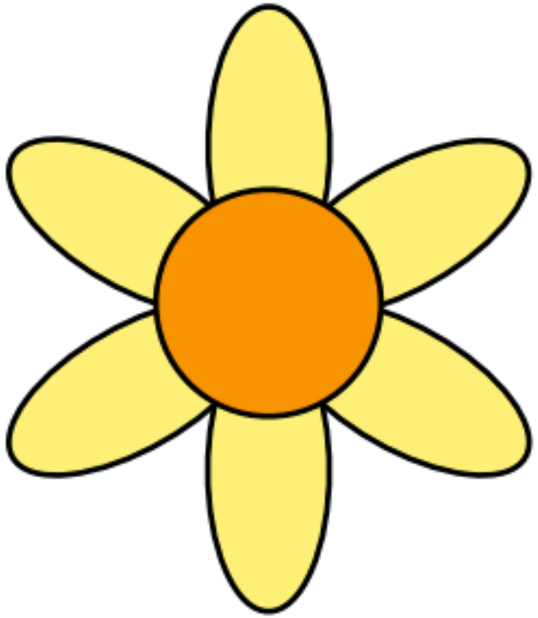
$$t(e) \geq |X \cap Y| \geq |X| + |Y| - |X \cup Y| \geq \deg(u) + \deg(v) - n,$$

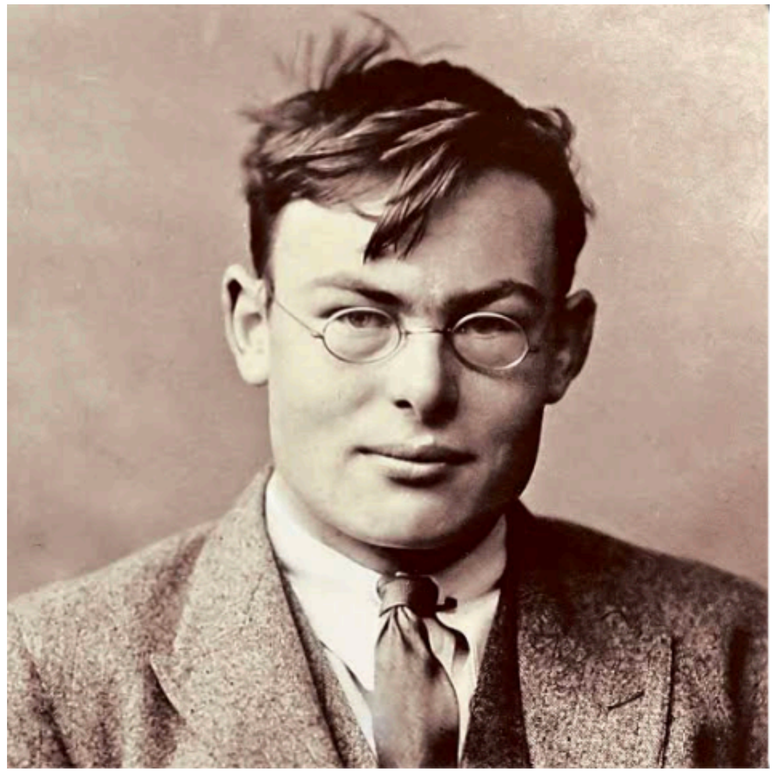
where $t(e)$ is the number of triangles containing e.

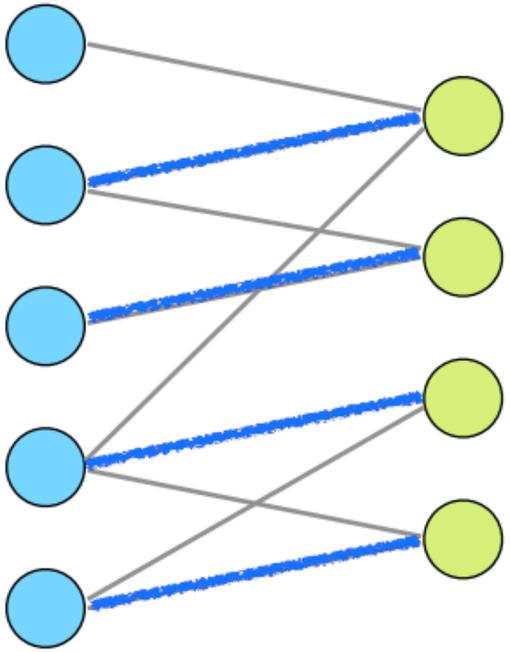
Then we have

$$t(G) \geq \frac{1}{3}\sum_{e\in E} t(e) = \frac{1}{3}\sum_{v\in V} (\deg(\mathrm{v}))^2 - \frac{n|E|}{3} \geq \frac{1}{3n}\left(\sum_{v\in V} \deg(\mathrm{v})\right)^2 - \frac{n|E|}{3} = \frac{|E|}{3n}\left(4|E| - n^2\right),$$

where the second inequality is by Cauchy-Schwarz inequality.







Problem Set 4

extremal set theory, Ramsey theory, matching theory

Problem 1

Recall that the smallest number $R(k, \ell)$ satisfying the condition in the Ramsey theory is called the **Ramsey number**. Prove that:

- $R(4, 3) \leq 9$. (Hint: Proof by contradiction. Color the edges of K_9 in red and blue, and assume that there are no red triangles and no blue 4-cliques. Try to determine the number of red and blue edges adjacent to each vertex.)
- $R(4, 4) \leq 18$.

1. Assume (for the sake of contradiction) that there exists an edge coloring of K_9 so that no red triangles and no blue 4-cliques exist.

   For each vertex v, consider the number of blue edges adjacent to it, call it $\deg_b(v)$.

   Assume (again, for the sake of contradiction) there exists some v that $\deg_b(v) \leq 4$. Then let S be the set of vertices of V excluding v and all vertices that share a blue edge with v. Then we have $|S| \geq 4$. Note that for any $u, w \in S$, if $c(u, w) =$ red, then (u, v, w) forms a red triangle, contradiction. Hence, $c(u, w) =$ blue for all $u, w \in S$, then S forms a blue 4-clique, contradiction. Therefore, we have $\deg_b(v) > 4$ for all vertices v.

   Assume (once again, for the sake of contradiction) there exists some v that $\deg_b(v) \geq 6$. Then let S be the set of vertices that share a blue edge with v. Then $|S| \geq 6$. Since $R(3,3) = 6$, there must either exist a red triangle or blue triangle inside S. Both cases immediately lead to a contradiction. Therefore, we have $\deg_b(v) < 6$ for all vertices v.

   Overall, we have $\deg_b(v) = 5$ for all vertices v, which is also impossible since the sum of $\deg_b(v)$ must be even according to the handshaking lemma. Therefore, we have proved $R(4,3) \leq 9$.

Problem 1

Recall that the smallest number $R(k, \ell)$ satisfying the condition in the Ramsey theory is called the **Ramsey number**. Prove that:

- $R(4, 3) \leq 9$. (Hint: Proof by contradiction. Color the edges of K_9 in red and blue, and assume that there are no red triangles and no blue 4-cliques. Try to determine the number of red and blue edges adjacent to each vertex.)
- $R(4, 4) \leq 18$.

2. Since $R(4,3) \leq 9$, we have $R(4,4) \leq R(3,4) + R(4,3) \leq 18$.

Problem 2

We color each non-empty subset of $[n] = \{1, 2, \ldots, n\}$ with one of the r colors in $[r]$. Show that for any finite r there is a finite N such that for all $n \geq N$, for any r-coloring of non-empty subsets of $[n]$, there always exist $1 \leq i < j < k \leq n$ such that the intervals $[i, j) = \{i, i+1, \ldots, j-1\}$, $[j, k) = \{j, j+1, \ldots, k-1\}$ and $[i, k) = \{i, i+1, \ldots, k-1\}$ are all assigned with the same color.

We show that for any finite r there is a finite N such that for all $n \geq N$, for any r-edge coloring of K_n there always exists a triangle with all three edges having the same color. This implies the given statement by considering an edge (i, j) as an interval $[i, j]$ in the original statement.

When $r = 2$, it suffices to take $N(r) = R(3,3) = 6$. We then apply induction and show that taking $N(r+1) = (r+1)N(r) + 1$ suffices.

Consider a vertex v and group all its neighbours by the color of the edge (there are $r + 1$ colors) with v. Then the largest set S satisfy $|S| \geq \dfrac{(r+1)N(r)}{(r+1)} = N(r)$. Let c be the color of the edge between v and any vertex in v. If any two vertices of S has edge of color c, we have found a monochromatic triangle. Otherwise, all edges in S can only take r colors, hence a monochromatic triangle must exist according to the induction hypothesis.

Problem 3

(Frankl 1986)

Let $\mathcal{F} \subseteq \binom{[n]}{k}$ be a k-uniform family, and suppose that it satisfies that $A \cap B \not\subset C$ for any $A, B, C \in \mathcal{F}$.

- Fix any $B \in \mathcal{F}$. Show that the family $\{A \cap B \mid A \in \mathcal{F}, A \neq B\}$ is an anti chain.
- Show that $|\mathcal{F}| \leq 1 + \binom{k}{\lfloor k/2 \rfloor}$.

1. This is simply by verifying the conditions. We therefore skip it.

2. Fix any $B \in \mathscr{F}$. Note that by (1.) and Sperner's theorem we have

$$|\{A \cap B \mid A \in \mathscr{F}, A \neq B\}| \leq \binom{k}{\lfloor \frac{k}{2} \rfloor}.$$

By the given condition we also have that $A \cap B \neq A' \cap B$ for all $A, A' \in \mathscr{F}, A, A' \neq B$. Hence

$$|\mathscr{F}| \leq 1 + |\{A \cap B \mid A \in \mathscr{F}, A \neq B\}| \leq 1 + \binom{k}{\lfloor \frac{k}{2} \rfloor}.$$

Problem 4

Suppose that M, M' are matchings in a bipartite graph G with bipartition A, B. Suppose that all the vertices of $S \subseteq A$ are matched by M and that all the vertices of $T \subseteq B$ are matched by M'. Prove that G contains a matching that matches all the vertices of $S \cup T$.

Problem 5

Use the **König-Egerváry theorem** to prove the followings:

- Every bipartite graph G with l edges has a matching of size at least $l/\Delta(G)$, where $\Delta(G)$ is the maximum degree of a vertex in G.
- Let A be a 0-1 matrix with m 1s. Let s be the maximal number of 1s in a row or column of A, and suppose that A has no square $r \times r$ all-1 sub-matrix. It requires at least $m/(sr)$ all-1 (not necessarily square) sub-matrices to cover all 1s in A.

Problem 6 (Bonus Problem)

This is a bonus problem worth 20 points, the score of which will be used to replace the lowest score of any other problem in all problem sets. (Nothing will be done if this problem is already the lowest-scored problem)

A company with n two-person teams researching products adapted to the pandemic by scheduling so no more than n individuals were in the office simultaneously, ensuring smooth operations even post-pandemic. They organized teams numbered 1 to n, with members labeled $(i, 1)$ and $(i, 2)$. Each week, one team member works onsite, while the other works remotely, maintaining productivity without in-person meetings between team members. The employees $(i, 1)$ and $(i, 2)$ know each other well and collaborate productively regardless of being isolated from each other, so members of the same team do not need to meet in person in the office. However, isolation between members from different teams is still a concern.

Each pair of teams i and j for $i \neq j$ has to collaborate occasionally. For a given number w of weeks and for fixed team members (i, a) and (j, b), let $w_1 < w_2 < \cdots < w_k$ be the weeks in which these two team members meet in the office. The *isolation* of those two people is the **maximum** of

$$\{w_1, w_2 - w_1, w_3 - w_2, \ldots, w_k - w_{k-1}, w + 1 - w_k\},$$

or infinity if those two people never meet. The isolation of the whole company is the **maximum** isolation across all choices of i, j, a and b.

Given n and w, let $f(n, w)$ be the **minimum** isolation over all possible schedules. For example, $f(2, 3) = \infty$ and $f(2, 6) = 4$. Give an expression for $f(n, w)$ and explain your answer. (Hint: View a schedule for one team as a binary string $\mathbf{x}$ of length w, with $\mathbf{x}_i = 0$ indicating that the first team member comes to work on day i, and $x_i = 1$ indicating that the second team member comes to work on day i. What is the criterion between any two binary strings if we need the isolation being at most k for some integer k?)

Remark. This problem comes from Problem B: Schedule, of the 47th ICPC World Finals. The two teaching assistants participated the contest in the same team, but failed to solve this problem in contest.

47th Annual ICPC World Championship | icpc.foundation | World Finals | ICPC 2023 Luxor | icpc International Collegiate Programming Contest | hosted by AASTMT

Problem B
Schedule
Time limit: 2 seconds

The Institute for Creative Product Combinations (ICPC) tries to find unusual and innovative ways to unite seemingly unrelated products or technologies, opening up new markets and creating new jobs. (For instance, their most recent success was the "hairbachi," a hair-dryer with a hibachi grill top attachment for preparing on-the-go hot meals.) The company employs n teams of size 2 to research individual products, then members of the different teams get together to explore ways of combining products.

During the pandemic, the ICPC management organized everyone's schedule in such a way that there were never more than n people in the office at the same time, and things ran so smoothly that they continued the process once things began to return to normal. Here is the scheme they used. Label the teams with integers 1 through n and the two people on the ith team as $(i, 1)$ and $(i, 2)$ for each i from 1 to n. Each week, exactly one person from each team is allowed in the office, while the other has to stay away. The employees $(i, 1)$ and $(i, 2)$ know each other well and collaborate productively regardless of being isolated from each other, so members of the same team do not need to meet in person in the office. However, isolation between members from different teams is still a concern.

Each pair of teams i and j for $i \neq j$ has to collaborate occasionally. For a given number w of weeks and for fixed team members (i, a) and (j, b), let $w_1 < w_2 < \ldots < w_k$ be the weeks in which these two team members meet in the office. The isolation of those two people is the maximum of

$$\{w_1, w_2 - w_1, w_3 - w_2, \ldots, w_k - w_{k-1}, w + 1 - w_k\},$$

Problem 6 (Bonus Problem)

This is a bonus problem worth 20 points, the score of which will be used to replace the lowest score of any other problem in all problem sets. (Nothing will be done if this problem is already the lowest-scored problem)

A company with n two-person teams researching products adapted to the pandemic by scheduling so no more than n individuals were in the office simultaneously, ensuring smooth operations even post-pandemic. They organized teams numbered 1 to n, with members labeled $(i, 1)$ and $(i, 2)$. Each week, one team member works onsite, while the other works remotely, maintaining productivity without in-person meetings between team members. The employees $(i, 1)$ and $(i, 2)$ know each other well and collaborate productively regardless of being isolated from each other, so members of the same team do not need to meet in person in the office. However, isolation between members from different teams is still a concern.

Each pair of teams i and j for $i \neq j$ has to collaborate occasionally. For a given number w of weeks and for fixed team members (i, a) and (j, b), let $w_1 < w_2 < \cdots < w_k$ be the weeks in which these two team members meet in the office. The *isolation* of those two people is the **maximum** of

$$\{w_1, w_2 - w_1, w_3 - w_2, \ldots, w_k - w_{k-1}, w + 1 - w_k\},$$

or infinity if those two people never meet. The isolation of the whole company is the **maximum** isolation across all choices of i, j, a and b.

Given n and w, let $f(n, w)$ be the **minimum** isolation over all possible schedules. For example, $f(2, 3) = \infty$ and $f(2, 6) = 4$. Give an expression for $f(n, w)$ and explain your answer. (Hint: View a schedule for one team as a binary string $\mathbf{x}$ of length w, with $\mathbf{x}_i = 0$ indicating that the first team member comes to work on day i, and $x_i = 1$ indicating that the second team member comes to work on day i. What is the criterion between any two binary strings if we need the isolation being at most k for some integer k?)

Suppose we have some schedule with isolation k. Then during the first k weeks, every pair of individuals on different teams meet at least once, and if we repeat the schedule for these first k weeks indefinitely, we get a periodic schedule for arbitrarily many weeks with the same separation. This means that the problem is equivalent to finding the smallest k such that all people can meet at least once in k weeks. If $k \leq w$ then we take that schedule and repeat it to get a full schedule of w weeks, and if $k > w$ then the answer is ∞.

Equivalently, we can fix some k and find the largest n.

We can view a k-day schedule for one team as a binary string $\mathbf{x}$ of length k, with $\mathbf{x}_i = 0$ indicating that the first team member comes to work on day i, and $\mathbf{x}_i = 1$ indicating that the second team member comes to work on day i. Two binary strings $\mathbf{x}$ and $\mathbf{y}$ are compatible if for all four combinations $c \in \{00,01,10,11\}$ there is some i such that $\mathbf{x}_i\mathbf{y}_i = c$. A schedule for n teams is then a list of n binary strings that are pairwise compatible.

Problem 6 (Bonus Problem)

This is a bonus problem worth 20 points, the score of which will be used to replace the lowest score of any other problem in all problem sets. (Nothing will be done if this problem is already the lowest-scored problem)

A company with n two-person teams researching products adapted to the pandemic by scheduling so no more than n individuals were in the office simultaneously, ensuring smooth operations even post-pandemic. They organized teams numbered 1 to n, with members labeled $(i, 1)$ and $(i, 2)$. Each week, one team member works onsite, while the other works remotely, maintaining productivity without in-person meetings between team members. The employees $(i, 1)$ and $(i, 2)$ know each other well and collaborate productively regardless of being isolated from each other, so members of the same team do not need to meet in person in the office. However, isolation between members from different teams is still a concern.

Each pair of teams i and j for $i \neq j$ has to collaborate occasionally. For a given number w of weeks and for fixed team members (i, a) and (j, b), let $w_1 < w_2 < \cdots < w_k$ be the weeks in which these two team members meet in the office. The *isolation* of those two people is the **maximum** of

$$\{w_1, w_2 - w_1, w_3 - w_2, \ldots, w_k - w_{k-1}, w + 1 - w_k\},$$

or infinity if those two people never meet. The isolation of the whole company is the **maximum** isolation across all choices of i, j, a and b.

Given n and w, let $f(n, w)$ be the **minimum** isolation over all possible schedules. For example, $f(2, 3) = \infty$ and $f(2, 6) = 4$. Give an expression for $f(n, w)$ and explain your answer. (Hint: View a schedule for one team as a binary string $\mathbf{x}$ of length w, with $\mathbf{x}_i = 0$ indicating that the first team member comes to work on day i, and $x_i = 1$ indicating that the second team member comes to work on day i. What is the criterion between any two binary strings if we need the isolation being at most k for some integer k?)

(Formalizing the problem: maximum intersecting Sperner family) without loss of generality we can assume that all strings in a schedule start with 0, because if we flip 0 and 1 in a string, it remains compatible with the same strings. Also we can consider each team as a subset of $[k-1]$, and all teams as a set family $\mathscr{F}$. Then we require that

(1) for each pair of teams $\mathbf{x}$ and $\mathbf{y}$ there must both be some i such that $\mathbf{x}_i\mathbf{y}_i = 01$ and some j such that $\mathbf{x}_j\mathbf{y}_j = 10$ ($\mathscr{F}$ is an antichain, i.e., $\mathscr{F}$ is a Sperner family).

(2) for each pair of teams $\mathbf{x}$ and $\mathbf{y}$ there must both be some i such that $\mathbf{x}_i\mathbf{y}_i = 1$ ($\mathscr{F}$ is intersecting).

What we are asking is the maximum size of $|\mathscr{F}|$.

Problem 6 (Bonus Problem)

This is a bonus problem worth 20 points, the score of which will be used to replace the lowest score of any other problem in all problem sets. (Nothing will be done if this problem is already the lowest-scored problem)

A company with n two-person teams researching products adapted to the pandemic by scheduling so no more than n individuals were in the office simultaneously, ensuring smooth operations even post-pandemic. They organized teams numbered 1 to n, with members labeled $(i, 1)$ and $(i, 2)$. Each week, one team member works onsite, while the other works remotely, maintaining productivity without in-person meetings between team members. The employees $(i, 1)$ and $(i, 2)$ know each other well and collaborate productively regardless of being isolated from each other, so members of the same team do not need to meet in person in the office. However, isolation between members from different teams is still a concern.

Each pair of teams i and j for $i \neq j$ has to collaborate occasionally. For a given number w of weeks and for fixed team members (i, a) and (j, b), let $w_1 < w_2 < \cdots < w_k$ be the weeks in which these two team members meet in the office. The *isolation* of those two people is the **maximum** of

$$\{w_1, w_2 - w_1, w_3 - w_2, \ldots, w_k - w_{k-1}, w + 1 - w_k\},$$

or infinity if those two people never meet. The isolation of the whole company is the **maximum** isolation across all choices of i, j, a and b.

Given n and w, let $f(n, w)$ be the **minimum** isolation over all possible schedules. For example, $f(2, 3) = \infty$ and $f(2, 6) = 4$. Give an expression for $f(n, w)$ and explain your answer. (Hint: View a schedule for one team as a binary string $\mathbf{x}$ of length w, with $\mathbf{x}_i = 0$ indicating that the first team member comes to work on day i, and $x_i = 1$ indicating that the second team member comes to work on day i. What is the criterion between any two binary strings if we need the isolation being at most k for some integer k?)

Claim: $|\mathscr{F}| \leq \binom{k-1}{\lceil \frac{k}{2} \rceil}$ and there exists $|\mathscr{F}| = \binom{k-1}{\lceil \frac{k}{2} \rceil}$ (A schedule exists if and only if $n \leq \binom{k-1}{\lceil \frac{k}{2} \rceil}$).

(Construction of $|\mathscr{F}| = \binom{k-1}{\lceil \frac{k}{2} \rceil}$) Simply let $\mathscr{F} = \binom{[k-1]}{\lceil \frac{k}{2} \rceil}$. It's easy to see that $\mathscr{F}$ is a Sperner family and that $\mathscr{F}$ is intersecting.

(Proof of $|\mathscr{F}| \leq \binom{k-1}{\lceil \frac{k}{2} \rceil}$) When k is even the proof is trivial by Sperner's theorem. We then assume k is odd. There are multiple proofs by adapting proof methods for Sperner's theorem and Erdős–Ko–Rado theorem. We present here an elegant proof given by Katona in 1997, which is similar to Lubell's proof for the Sperner's theorem.

Claim: Let $\mathscr{C}$ be a cyclic permutation of $[k-1]$. Let $B_1, B_2, \ldots, B_r$ be a Sperner family of intervals in $\mathscr{C}$. Then

$$\sum_{i=1}^{r} \binom{k-1}{|B_i|} \leq (k-1)\binom{k-1}{\lceil \frac{k}{2} \rceil}.$$

Proof: Consider the intervals starting from a given element to the right along $\mathscr{C}$. Obviously at most one of them can be a B_i , since the family is Sperner. Therefore $r \leq k-1$ holds. This settles the proof if n is odd, so let us suppose that n is even. Two cases will be distinguished.

$r = k-1$:Suppose that B_i starts at the i-th element along $\mathscr{C}$. The Sperner property implies $|B_i| \leq |B_{i+1}|$. Hence $|B_1| \leq |B_2| \leq \ldots \leq |B_r| \leq |B_1|$, and all sets have the same size. The size cannot be $\dfrac{k-1}{2}$ because of the intersecting property, finishing the proof of this case.

$r < k-1$:At most one of the complementing $\dfrac{k-1}{2}$-element intervals can occur among the Bs, therefore at most $\dfrac{k-1}{2}$ of the terms on the LHS can be $\binom{k-1}{\frac{k-1}{2}}$, the others cannot exceed $\binom{k-1}{\lceil \frac{k}{2} \rceil}$.

Summing these upper bounds yields the RHS.

(Proof of $|\mathscr{F}| \leq \binom{k-1}{\lceil \frac{k}{2} \rceil}$, cont'd) We then consider $\sum_{\mathscr{C}, S \in \mathscr{F}} \binom{k-1}{|S|}$ for those pairs where S is an interval along $\mathscr{C}$. It is easy to see that each $S \in \mathscr{F}$ is an interval in exactly $|S|!(k-1-|S|)!$ cyclic permutations of $[k-1]$. Hence,

$$\sum_{S \in \mathscr{F}} \sum_{\mathscr{C} : S \text{ is an interval}} \binom{k-1}{|S|} = \sum_{S \in \mathscr{F}} |S|!(k-1-|S|)! \binom{k-1}{|S|} = (k-1)!\,|\mathscr{F}|\,.$$

Note that the previous claim gives an upper bound that

$$\sum_{S \in \mathscr{F}} \sum_{\mathscr{C} : S \text{ is an interval}} \binom{k-1}{|S|} \leq (k-2)! \cdot (k-1) \cdot \binom{k-1}{\lceil \frac{k}{2} \rceil}.$$

Combining the two finishes the proof.