# 1 Pledge

Berkeley Honor Code: As a member of the UC Berkeley community, I act with honesty, integrity, and respect for others.

In particular, I acknowledge that:

- I alone am taking this exam. Other than with the instructor and GSI, I will not have any verbal, written, or electronic communication about the exam with anyone else while I am taking the exam or while others are taking the exam.
- I will not have any other browsers open while taking the exam.
- I will not refer to any books, notes, or online sources of information while taking the exam, other than what the instructor has allowed.
- I will not take screenshots, photos, or otherwise make copies of exam questions to share with others.

SIGN Your Name: ________________________

*(Rewritten by Alec Li, as the original solutions have mistakes and unclear answers.)*

## 2 Warmup: Shoeless

There are 70 people. An even number of them wear one shoe. One half of the remainder wear no shoes, and the rest wear two shoes.

1. How many shoes are worn in total by the 70 people?

70

2. Briefly justify your answer.

**Answer:** We know that some number of people are wearing one shoe. From the remainder, we can think of each person with two shoes as giving one of their shoes to someone wearing no shoes. Since there are an equal number of people wearing no shoes compared to people wearing two shoes, at the end of this process everybody will be wearing one shoe. This means that the number of shoes is equal to the number of people, i.e. there are 70 shoes.

Alternatively, we can set up an equation; let $x$ be the number of people wearing one shoe. The number of shoes is then

$$x + \frac{1}{2}(70 - x) \cdot 2 = x + 70 - x = 70.$$

## 3 Propositional Logic

Let $A$, $B$, and $C$ be statements, where $A$ is always true, $B$ is always false, $C$ is always true, and let $P(x)$ and $Q(x)$ be predicates over a nonempty set $S$, where the statements $\forall x \in S,\ P(x)$; $\exists x \in S,\ Q(x)$; and $\exists x \in S,\ \neg Q(x)$ are always true.

1. $A \implies B$

○ Always True ● **Always False** ○ Depends

**Answer:** This implication simplifies to "True $\implies$ False", which is false by the definition of implications.

2. $B \implies A$

● **Always True** ○ Always False ○ Depends

**Answer:** This implication simplifies to "False $\implies$ True", which is true by the definition of implications.

3. $A \implies C$

● **Always True** ○ Always False ○ Depends

**Answer:** This implication simplifies to "True $\implies$ True", which is true by the definition of implications.

4. $(A \vee B) \implies \neg C$

○ Always True ● **Always False** ○ Depends

**Answer:** Directly plugging in, we have "((True) $\vee$ (False)) $\implies$ $\neg$(True)". The LHS simplifies to True, and the RHS simplifies to False, meaning the implication simplifies to "True $\implies$ False", which is false.

5. $\neg A \vee C$

● **Always True** ○ Always False ○ Depends

**Answer:** We can directly plug in to get "¬(True) ∨ (True)", which evaluates to true.

Alternatively, we can see that $\neg A \lor C \equiv A \implies C$, which is true from earlier.

6. $\forall x \in S,\ Q(x)$

○ Always True ● **Always False** ○ Depends

**Answer:** We're given that $\exists x \in S, \neg Q(x)$, so $Q(x)$ can't possibly be true for all $x \in S$.

7. $\exists x \in S,\ P(x)$

● **Always True** ○ Always False ○ Depends

**Answer:** We're given that $\forall x \in S,\ P(x)$, meaning $P(x)$ is true for all $x$, so there must be some $x$ that makes $P(x)$ true (any of them).

8. $\exists x \in S,\ Q(x) \land P(x)$

● **Always True** ○ Always False ○ Depends

**Answer:** We know that there exists an $x$ that makes $Q(x)$ true, and we also know that $P(x)$ is true for all $x$, including the $x$ that makes $Q(x)$ true. As such, there does exist an $x$ that makes both $P(x)$ and $Q(x)$ true.

9. $\forall x \in S,\ Q(x) \lor P(x)$

● **Always True** ○ Always False ○ Depends

**Answer:** Since we know $P(x)$ is true for all $x$, it doesn't matter what $Q(x)$ is, as $P(x)$ is always true for all $x \in S$, making the conjunction true.

## 4 To Prove or Disprove, that is the question

First, say whether each statement is true or false, and then prove or disprove. The proof or counterexamples should be brief.

1. Prove or disprove: for $x, y, d \in \mathbb{Z}$, if $d \mid (x-y)$, then $d \mid x$.

○ True ● **False**

**Answer:** As a counterexample, we can let $x = y$ and $d = x+1$. In this case, $d \mid (x-y)$ since $d \mid 0$ (anything divides 0), but $(x+1) \nmid x$.

For a more concrete example, consider $x = y = 2$ and $d = 3$. We have $3 \mid (2-2)$, but $3 \nmid 2$.

2. Prove or disprove: $2x^2 = 4$ has no solutions in the rationals.

● **True** ○ False

**Answer:** Suppose for contradiction that there is a rational solution to $2x^2 = 4$. This means that there is a rational solution to $x^2 = 2$, with $x = \sqrt{2}$. However, we know from lecture and the notes that $\sqrt{2}$ is irrational, or equivalently there is no rational solution to $x^2 = 2$; this is a contradiction.

This means that there must not be a rational solution to $2x^2 = 4$.

## 5 Stability in Matchings

1. For any stable matching instance, the job optimal stable matching has at least one job that is paired with their favorite candidate.

○ True ● **False**

**Answer:** Consider the following stable matching instance:

| Job | Candidates |
|---|---|
| $A$ | $1>2>3$ |
| $B$ | $1>2>3$ |
| $C$ | $2>1>3$ |

| Candidate | Jobs |
|---|---|
| 1 | $C>A>B$ |
| 2 | $A>B>C$ |
| 3 | $A>C>B$ |

Running the propose and reject algorithm (which gives the job optimal stable matching), we have

| Candidate | Day 1 | Day 2 | Day 3 | Day 4 | Day 5 |
|---|---|---|---|---|---|
| 1 | $A,\cancel{B}$ | $A$ | $\cancel{A},C$ | $C$ | $C$ |
| 2 | $C$ | $B,\cancel{C}$ | $B$ | $A,\cancel{B}$ | $A$ |
| 3 | – | – | – | – | $B$ |

This gives the matching $(A,2)$, $(B,3)$, $(C,1)$, which is a job optimal matching none of the jobs get their favorite candidate.

2. For any stable matching instance, the job optimal stable matching has no job paired with their least favorite candidate.

○ True ● **False**

**Answer:** Consider the same instance from the previous question. Here, we have $B$ matched with its least favorite candidate, 3.

In general, if all jobs have a common least favorite candidate, *some* job must end up paired with it in the end.

3. For any stable matching instance, the job optimal stable matching has at least one candidate that does not get their favorite job.

○ True ● **False**

**Answer:** Consider the following stable matching instance:

| Job | Candidates |
|---|---|
| $A$ | $1>2$ |
| $B$ | $2>1$ |

| Candidate | Jobs |
|---|---|
| 1 | $A>B$ |
| 2 | $B>A$ |

Here, the job optimal stable matching is $\{(A,1),(B,2)\}$, where all jobs and candidates get their favorite choices.

In general, if every job has a distinct favorite candidate, and each one of those candidates like the corresponding job the most, then the propose and reject algorithm ends in one day, and every job and every candidate gets their favorite choices.

4. For any stable matching instance, all matchings have an even number of rogue couples. (Recall, a stable matching has 0 rogue couples.)

○ True ● **False**

**Answer:** Consider the following stable matching instance:

| Job | Candidates |
|---|---|
| $A$ | $1>2$ |
| $B$ | $2>1$ |

| Candidate | Jobs |
|---|---|
| 1 | $A>B$ |
| 2 | $A>B$ |

Consider the matching $\{(A,2),(B,1)\}$. Here, $(A,1)$ is a rogue couple, but $B$ is not in a rogue couple, since nobody likes $B$. This means that this matching has exactly one rogue couple, which is not even.

5. Consider an output from running the Propose-and-Reject algorithm on a stable matching instance with $n$ jobs and $n$ candidates. We then arbitrarily permute one job's preference list.

(a) What is the maximum number of jobs that can participate in a rogue couple in the outputted matching with respect to the permuted preference list?

1

**Answer:** The only job that can participate in a rogue couple is the one whose preference list was permuted.

To see why, suppose we look at some other job $j$, paired with candidate $c$ in the matching. All candidates that $j$ prefers more than $c$ would have rejected $j$ for some other job they liked more. This means that $j$ can't be in a rogue couple with any other candidate—the other candidate wouldn't like $j$ more than what they currently have.

(b) What is the maximum number of rogue couples in the outputted matching with respect to the permuted preference list?

$n-1$

**Answer:** We know that no other job can be in a rogue couple, so all possible rogue couples must include the job whose preference list we permuted.

Suppose the job whose preference list we permuted is job $j$, paired with candidate $c$ in the matching. If all other candidates put $j$ at the top of their preference lists, and the permutation moved $c$ to the bottom of $j$'s preference list, then $(j, c')$ will be a rogue couple for all other candidates $c'$. This is because $j$ would prefer any other candidate $c'$ more than $c$, and any other candidate $c'$ prefers $j$ the most.

Since there are $n-1$ other candidates to form the rogue couple with, the maximum number of rogue couples with respect to the permuted preference list is $n-1$.

## 6 Graphs

You may assume all graphs in this section are simple (as defined in the notes) unless otherwise specified.

1. For all $n \geq 3$, any connected graph with $n$ vertices and $n$ edges is planar.

● **True** ○ False

**Answer:** A graph with $n$ vertices and $n$ edges is a tree with an additional edge. All trees are planar, and adding an edge will never cause a crossing; there is always a path between any two vertices, since a tree has only one face—there won't ever be an edge blocking the path.

2. How many colors are needed to vertex color a bipartite graph of maximum vertex degree $d$? (Recall that a valid vertex coloring assigns colors to the vertices such that the vertices in an edge have different colors.)

2

**Answer:** Any bipartite graph is 2-colorable; we can color each group with one color. All edges are between these two groups, so this is always a valid coloring.

3. Consider that $G = (V, E_1)$ and $G' = (V, E_2)$ are bipartite, how many colors are sufficient to vertex color $G'' = (V, E_1 \cup E_2)$? (You should give as small a bound as possible.)

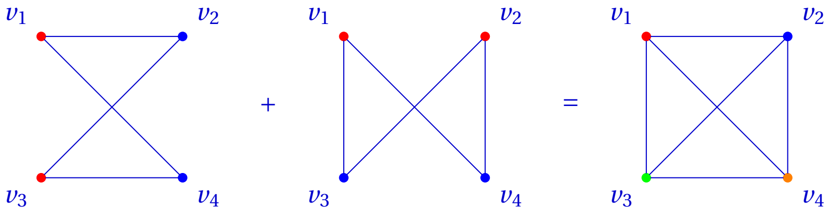
   4

   **Answer:** We can assign a bitstring to each vertex, representing the color for each vertex. In particular, $G$ can be colored with two colors; let's say colors 0 and 1. Graph $G'$ can also be colored similarly, with colors 0 and 1.

   In the combined graph $(V, E_1 \cup E_2)$, we can color a vertex with a length 2 bitstring. The first digit corresponds to its color in $G$, and the second digit corresponds to its color in $G'$. For example, a vertex colored with 0 in $G$ and colored with 1 in $G'$ will be colored with 01 in $G''$.

   Here, we need 4 colors (one for each possible length-2 bitstring) to color $G''$. We can also see that the coloring is valid: for any edge $(u, v)$, if it came from $G$, then the first digits of the colors of $u$ and $v$ will differ, and if it came from $G'$, then the second digits of the colors of $u$ and $v$ will differ. This means that $u$ and $v$ will always be colored with different colors, no matter which edge we look at, making this a valid coloring.

   Further, we can see that 4 colors is the tightest bound possible; consider the graph on 4 vertices $V = \{v_1, v_2, v_3, v_4\}$. With $E_1 = \{(v_1, v_2), (v_3, v_4), (v_1, v_4), (v_2, v_3)\}$ and $E_2 = \{(v_1, v_3), (v_2, v_4), (v_1, v4), (v_2, v_3)\}$, the resulting graph $G'' = (V, E_1 \cup E_2)$ is equivalent to $K_4$, which requires 4 colors to vertex color. An image is shown below:



4. Consider bipartite graphs $(V, E_1)$, $(V, E_2)$, $(V, E_3)$, …, $(V, E_k)$ are bipartite, how many colors are sufficient to color $(V, E_1 \cup \cdots \cup E_k)$? (You should give as small a bound as possible that is in terms of $k$.)

   $2^k$

   **Answer:** We can generalize the previous part to a $k$-digit coloring with length $k$ bitstrings. This corresponds to $2^k$ colors, and for any given edge, if it comes from $E_i$, the $i$th digit will be different in the colors of its adjacent vertices.

   We can also show that $2^k$ is a tight bound; consider a vertex set $V$ with $2^k$ vertices, each labeled with a different length $k$ bitstring. We can then construct the graph $(V, E_i)$ by looking at the $i$th digit in the bitstring for each digit. If the $i$th digit is a 0, we put it in group 0, and if the $i$th digit is a 1, we put it in group 1. For each of these graphs, we include all possible edges between the vertices (i.e. each graph will be a complete bipartite graph).

   If we look at the union $(V, E_1 \cup E_2 \cup \cdots \cup E_k)$, we claim that this forms the complete graph on $2^k$ vertices, $K_{2^k}$. In particular, we know that there is a vertex between two vertices if any digit differs between them; if the $i$th digit is different, then $E_i$ will have an edge between the two vertices.

   Since all vertices have different bitstrings, there will be an edge between all pairs of vertices, and this creates $K_{2^k}$, which requires $2^k$ colors to vertex color.

5. There is always a vertex of degree at most ________________ in a connected bipartite planar graph. Recall that any bipartite planar graph $G = (V, E)$ satisfies $|E| \le 2|V| - 4$. (You should give as tight of a bound as possible.)

$\boxed{3}$

**Answer:** The total degree is equal to $2|E|$ by the handshaking lemma. The inequality given in the question can be multiplied by 2, which gives $2|E| \leq 4|V| - 8$, meaning the total degree is at most $4|V| - 8$. Dividing by $|V|$, we have an average degree of at most $4 - \frac{8}{|V|}$, i.e. the average degree is always strictly less than 4.

This means that there will always be a vertex of degree at most 3 in the graph to bring down the average degree to below 4.

6. The complete graph $K_n$ on $n > 3$ vertices can be made to contain an Eulerian tour by deleting a minimum of ________________ edges. (Answer(s) should be as small as possible and possibly in terms of $n$.)

even $n$: $\boxed{\frac{n}{2}}$

odd $n$: $\boxed{0}$

**Answer:** If $n$ is even, then every vertex has an odd degree (each vertex is connected to $n-1$ vertices, which is odd since $n$ is even). This means that we can remove $\frac{n}{2}$ edges, each connected to pairs of distinct vertices, so that we decrease the degree of every vertex by 1. After the removal of these edges, now every vertex has even degree, and there exists a Eulerian tour.

If $n$ is odd, then every vertex has an even degree (each vertex is connected to $n-1$ vertices, which is even since $n$ is odd). This means that a Eulerian tour already exists, and we don't need to remove any edges.

7. Consider a connected $n$-vertex graph $G$ with exactly $k$ cycles. Provide as tight of a bound as possible for each part. (You may assume $n > 4k$.)

   (a) Remove $2k$ edges from $G$ produces a graph with at least ________________ connected components.

   $\boxed{k+1}$

   **Answer:** If we want to avoid creating new connected components, the first $k$ edges in the best case will remove all $k$ cycles, turning the graph into a tree. Each additional edge removed will increase the number of connected components by 1, going from 1 connected components to $k+1$ connected components.

   Intuitively, this is because removing an edge from a tree splits it into two connected components, each component being its own tree; any further edges removed will remove an edge from one of these smaller trees, where the same reasoning applies.

   (b) Removing $2k$ edges from $G$ produces a graph with at most ________________ connected components.

   $\boxed{2k+1}$

**Answer:** If we want to try to create new connected components, all $2k$ edges can create a new connected component if none of the edges we remove are in a cycle. This increases the number of connected components from 1 connected component to $2k+1$ connected components.

8. At least $d$ colors are *required* for a valid vertex coloring for any graph with maximum vertex degree $d$.

○ True ● **False**

**Answer:** Consider a complete bipartite graph on two vertex sets of size $d$. Each vertex has degree $d$, but the graph can still be vertex colored with 2 colors (one for each set).

9. Removing any degree 2 vertex (and its incident edges) in a connected acyclic graph leaves a graph with two connected components.

● **True** ○ False

**Answer:** It's most intuitive if you draw this out; removing the degree 2 vertex will always cut the graph into two connected components, as there is no way to get from one side of the removed vertex to the other.

Formally, we know that the original graph is a tree (as it is connected and acyclic), so we originally had $n$ vertices and $n-1$ edges. Because of this, removing the degree 2 vertex and its incident edges creates a graph with $n-1$ vertices and $n-3$ edges. Since there are no cycles in the graph (and will still have no cycles after removing a vertex and two edges), this final graph consists of connected components that are each also trees.

Suppose we have $k$ connected components in the final graph. Looking at the number of edges in total, we have $\sum_{i=1}^{k}(n_i-1)$ edges, where $n_i$ denotes the number of vertices in the $i$th connected component (here, we have $n_i-1$ edges per connected component, because each component is a tree, and thus has one less edge than the number of vertices in the component).

This sum in the final graph simplifies to $\left(\sum_{i=1}^{k} n_i\right)-k=(n-1)-k$; this is because the first quantity $\sum_{i=1}^{k} n_i$ sums over all vertex counts for each connected component, and we have a total of $n-1$ vertices in the final graph. We've calculated that there are $n-3$ edges in this graph, so we must have $k=2$ connected components.

10. Consider a walk in a connected graph $G=(V,E)$ with $|V|\geq 4$ formed by starting at a vertex $u$ and proceeding by choosing an arbitrary unused edge at the current vertex to get to the next vertex. The process terminates when it reaches a vertex where all incident edges have already been used.

(a) The walk always terminates at the vertex $u$ if and only if the degree of every vertex is ________________.

even

**Answer:** It's always possible to get stuck at a vertex of odd degree if one exists, since we must enter and leave vertices in pairs. This means that we always use up edges in pairs, so for vertices of odd degree, the last time we enter the vertex, there will be no corresponding edge to leave from, so we get stuck.

Because of this, there must not be any vertices of odd degree in the graph if we're guaranteed to terminate at vertex $u$; all vertices must have even degree.

(b) For a tree, the walk always terminates at a vertex with degree that is ________________. (Give as specific of an answer as possible.)

1

**Answer:** Since trees are acyclic, the walk will always end at a leaf. In particular, we can never return to a vertex we visited before (otherwise, we'd have found a cycle, and the graph wouldn't be a tree), so we'll keep visiting new vertices until we have no more adjacent vertices to visit. This will only occur at leaves of the tree.

(c) For a complete graph on $n$ vertices where $n$ is odd, the walk always forms a Hamiltonian tour.

○ True ● **False**

**Answer:** For $n \geq 4$, each vertex has degree at least 3. This means that we must visit some vertex at least twice to get stuck somewhere—we'd need to use up all of these incident edges in order to terminate, and to use up 3 edges we'd need to enter and leave the vertex, and then enter the vertex for a second time. This repetition means that the walk can't be a Hamiltonian tour.

(d) For a hypercube of dimension $n$, the walk terminates at an odd degree vertex or at $u$.

● **True** ○ False

**Answer:** This is true for any graph, not just for hypercubes.

If there exists an odd degree vertex, it is possible for the walk to terminate there, for reasons mentioned prior (we enter/leave vertices in pairs of edges, so we have one left over for an odd degree vertex, which stops us when we visit the vertex for the last time).

If there does not exist any odd degree vertices, then $u$ has an even degree, and the first edge in the walk leaves an odd number of edges remaining incident to $u$. Notice that we can't get stuck at any other vertices (again, because we enter/leave vertices in pairs), so we must end up back at $u$, where we terminate the walk—the last edge in the walk pairs with the first edge in the walk.

## 7 Modular Arithmetic: what number (or expression)?

1. Give all the solutions to $5x \equiv 3 \pmod{24}$ or write "none".

$$\boxed{15 \pmod{24}}$$

**Answer:** The multiplicative inverse $5^{-1} \equiv 5 \pmod{24}$, since $5 \cdot 5 = 25 \equiv 1 \pmod{24}$. This means that we have

$$\begin{aligned} 5x &\equiv 3 \pmod{24} \\ x &\equiv 5^{-1} \cdot 3 \pmod{24} \\ &\equiv 5 \cdot 3 \equiv 15 \pmod{24} \end{aligned}$$

2. Give all the solutions to $15x \equiv 3 \pmod{24}$ or write "none".

$$\boxed{5 \pmod{8}}$$

**Answer:** We can't solve this equation in the same way as the previous part, since $\gcd(15, 24) = 3 \neq 1$, and as such 15 has no inverse mod 24. However, we can convert this into an equation: $15x = 3 + 24k$ for some $k \in \mathbb{Z}$. Here, we can divide by 3 to get $5x = 1 + 8k$; converting back to an equivalence, we have $5x \equiv 1 \pmod{8}$.

Now, we can solve for $x$ using the fact that $5^{-1} \equiv 5 \pmod 8$:

$$\begin{aligned} 5x &\equiv 1 \pmod 8 \\ x &\equiv 5^{-1} \cdot 1 \pmod 8 \\ &\equiv 5 \cdot 1 = 5 \pmod 8 \end{aligned}$$

3. Give all the solutions to $15x \equiv 13 \pmod{24}$ or write “none”.

None

**Answer:** The LHS of the equivalence is a multiple of 3, but the RHS is not a multiple of 3. Adding or subtracting 24 to the RHS will never make it a multiple of 3, as 24 itself is a multiple of 3.

In particular, we have $15x = 13 + 24k$ for some $k \in \mathbb{Z}$. No matter which $k$ we choose, the RHS will never be a multiple of 3, while the LHS will always be a multiple of 3. This mismatch means that there is no solution to this equivalence.

4. Compute $21^{141} \pmod{71}$.

$21 \pmod{71}$

**Answer:** Notice that 71 is prime. This means that we can use FLT, which says that $21^{70} \equiv 1 \pmod{71}$, so $21^{141} = (21^{70})^2 \cdot 21 \equiv 21 \pmod{71}$.

5. Consider an RSA scheme with public key $N = 77$ and $e = 7$.

   (a) What is the private key?

   43

   **Answer:** Since $N = 77 = 7 \cdot 11$, we have $p = 11$ and $q = 7$, with $(p-1)(q-1) = 10 \cdot 6 = 60$. The private key is $d \equiv e^{-1} \pmod{(p-1)(q-1)}$, and in this case we want to find $d \equiv 7^{-1} \pmod{60}$. We can use the extended Euclidean algorithm to find this inverse.

   Doing this iteratively:

$$\begin{aligned} 60 &= 1 \cdot \mathbf{60} + 0 \cdot \mathbf{7} && (E_1) \\ 7 &= 0 \cdot \mathbf{60} + 1 \cdot \mathbf{7} && (E_2) \\ 4 &= 1 \cdot \mathbf{60} - 8 \cdot \mathbf{7} && (E_3 = E_1 - 8 \cdot E_2) \\ 3 &= -1 \cdot \mathbf{60} + 9 \cdot \mathbf{7} && (E_4 = E_2 - E_3) \\ 1 &= 2 \cdot \mathbf{60} - 17 \cdot \mathbf{7} && (E_5 = E_3 - E_4) \end{aligned}$$

   Under mod 60, we have $2 \cdot 60 - 17 \cdot 7 \equiv -17 \cdot 7 \equiv 1 \pmod{60}$. This means that $7^{-1} \equiv -17 \equiv 43 \pmod{60}$.

Alternatively, we can do this recursively. In the forward pass, we have

$$\begin{aligned}\gcd(60,7) &= \gcd(7, 60 \bmod 7) & \\ &= \gcd(7,4) & 4 &= \mathbf{60} - 8 \cdot \mathbf{7} \\ &= \gcd(4, 7 \bmod 4) & \\ &= \gcd(4,3) & 3 &= \mathbf{7} - 1 \cdot \mathbf{4} \\ &= \gcd(3, 4 \bmod 3) & \\ &= \gcd(3,1) & 1 &= \mathbf{4} - 1 \cdot \mathbf{3} \\ &= \gcd(1, 3 \bmod 1) & \\ &= \gcd(1,0) = 1 & \end{aligned}$$

In the backward pass, we have:

$$\begin{aligned}1 &= \mathbf{4} - 1 \cdot \mathbf{3} \\ &= \mathbf{4} - 1 \cdot (\mathbf{7} - 1 \cdot \mathbf{4}) \\ &= 2 \cdot \mathbf{4} - 1 \cdot \mathbf{7} \\ &= 2 \cdot (\mathbf{60} - 8 \cdot \mathbf{7}) - 1 \cdot \mathbf{7} \\ &= 2 \cdot \mathbf{60} - 17 \cdot \mathbf{7}\end{aligned}$$

This gives the same result, with $7^{-1} \equiv -17 \equiv 43 \pmod{60}$.

(b) What is the decoding of the encrypted message 76? (Give an answer in $\{0, \dots, 76\}$. Also, notice that 76 is one less than 77.)

$$\boxed{76 \pmod{77}}$$

**Answer:** Recall that to decrypt a ciphertext $x$, we have $D(x) = x^d$ where $d$ is the private key. Here, we have a private key of $d = 43$, and since $76 \equiv -1 \pmod{77}$, we have

$$76^{43} \equiv (-1)^{43} = -1 \equiv 76 \pmod{77}.$$

6. What is $304^{2022} \pmod{70}$? (Answer should be from $\{0, \dots, 69\}$.) Please clearly circle/box your final answer. (Hint: $70 = 2 \times 5 \times 7$.)

**Answer:** We can utilize the Chinese Remainder Theorem; splitting the mod into three parts, we can compute $x = 304^{2022}$ under mod 2, 5, and 7.

Since $304^{2022}$ is even, we have $x = 304^{2022} \equiv 0 \pmod{2}$.

We also have $304^{2022} \equiv (-1)^{2022} = 1 \pmod{5}$.

Lastly, we have $304^{2022} \equiv 3^{2022} \pmod{7}$. Simplifying this with FLT, we know that $3^6 \equiv 1 \pmod{7}$, and since $2022 = 6 \cdot 337$, we know that $3^{2022} = \left(3^6\right)^{337} \equiv 1 \pmod{7}$.

Together, we have the following system of equivalences:

$$\begin{cases} x \equiv 0 \pmod{2} \\ x \equiv 1 \pmod{5} \\ x \equiv 1 \pmod{7} \end{cases}.$$

We can solve this mechanically with CRT, or we can deduce the result from the system. Notice that the last two equivalences imply that $x \equiv 1 \pmod{35}$, and the first equivalence says that $x$ must be even. Under mod

70, we only have two choices; $x \equiv 1 \pmod{70}$ or $x \equiv 36 \pmod{70}$. Only the latter is even, so this means that $x \equiv \boxed{36 \pmod{70}}$ is our solution.

For completeness, here are the mechanical calculations for CRT:

- We want $\begin{cases} b_1 \equiv 0 \pmod{2} \\ b_1 \equiv 0 \pmod{5} \\ b_1 \equiv 0 \pmod{7} \end{cases}$

  Letting $b_1 = 0$ satisfies all of these equivalences.

- We want $\begin{cases} b_2 \equiv 0 \pmod{2} \\ b_2 \equiv 1 \pmod{5} \\ b_2 \equiv 0 \pmod{7} \end{cases}$

  Starting with $2 \cdot 7 = 14$ to satisfy the first and last equivalences, we want to find a $k$ such that $14k \equiv 1 \pmod{5}$, or $k \equiv 14^{-1} \equiv 4^{-1} \equiv 4 \pmod{5}$. This gives us $b_2 = 14 \cdot 4 = 56$.

- We want $\begin{cases} b_3 \equiv 0 \pmod{2} \\ b_3 \equiv 0 \pmod{5} \\ b_3 \equiv 1 \pmod{7} \end{cases}$

  Starting with $2 \cdot 5 = 10$ to satisfy the first two equivalences, we want to find a $k$ such that $10k \equiv 1 \pmod{7}$, or $k \equiv 10^{-1} \equiv 3^{-1} \equiv 5 \pmod{7}$. This gives us $b_3 = 10 \cdot 5 = 50$.

Together, we have $x = b_1 + b_2 + b_3 = 0 + 56 + 50 = 106 \equiv \boxed{36 \pmod{70}}$ as our solution.

## 8 When the Midterm has Proofs

Given a positive integer $n$, we define the digital root of $n$, DR($n$), to be the positive integer attained from repeatedly summing the base 10 digits of $n$ until $n$ is a single digit number. For example, $\text{DR}(191) = 2$ because $191 \to 1 + 9 + 1 = 11 \to 1 = 1 = 2$.

Prove that $\text{DR}(n) \equiv n \pmod{9}$.

**Answer:** We proceed by strong induction on $n$.

**Base Case** ($1 \le n \le 9$): If $n$ is a single digit, then $\text{DR}(n) = n$ and thus $\text{DR}(n) \equiv n \pmod{9}$.

**Induction Hypothesis**: Suppose $\text{DR}(k) \equiv k \pmod{9}$ for all $1 \le k \le n$.

**Inductive Step**: We will show the claim for $n = k + 1$. We can write $k + 1$ explicitly in base 10 as

$$k + 1 = a_0 10^0 + a_1 10^1 + \cdots + a_j 10^j,$$

for some $j \ge 1$ and $1 \le a_0, a_1, \ldots, a_j \le 9$ (here, $j$ is the number of decimal digits of $k + 1$).

Notice that $\text{DR}(k+1) = \text{DR}(a_0 + a_1 + \cdots + a_j)$, since we'd want to find the digital root of the sum of the digits of $k + 1$ in order to find the digital root of $k + 1$.

By the inductive hypothesis, since $1 \le a_0 + a_1 + \cdots + a_j \le k$, we know that $\text{DR}(a_0 + a_1 + \cdots + a_j) \equiv a_0 + a_1 + \cdots + a_j \pmod{9}$.

Equivalently, we know that

$$\begin{aligned} k + 1 &= a_0 10^0 + a_1 10^1 + \cdots + a_j 10^j \\ &\equiv a_0 1^0 + a_1 1^1 + \cdots + a_j 1^j \pmod{9} \\ &= a_0 + a_1 + \cdots + a_j \pmod{9} \end{aligned}$$

Together, we have

$$k+1 \equiv a_0 + a_1 + \cdots + a_j \equiv \mathrm{DR}(a_0 + a_1 + \cdots + a_j) = \mathrm{DR}(k+1) \pmod{9},$$

which proves the claim for $n = k+1$. By the principles of induction, the claim holds for all positive integers $n$.

## 9 Polynomial and Applications

1. Consider non-zero polynomials $P(x)$ of degree $d_p$ and $Q(x)$ of degree $d_q$, where $d_p$ and $d_q$ are nonnegative integers.

   (a) What is the maximum degree of $P(x)Q(x)$?

   $\boxed{d_q + d_p}$

   **Answer:** The exponent of the leading term of $P(x)Q(x)$ corresponds to the product of the leading terms in $P$ and $Q$, which have exponents equal to their degrees. This means that the maximum degree of $P(x)Q(x)$ is $d_q + d_p$, since multiplying the terms adds the exponents.

   (b) What is the maximum degree of $P(Q(x))$?

   $\boxed{d_p d_q}$

   **Answer:** Consider the leading term of $P$, $x^{d_p}$. If we plug in $x = Q(x)$, we have $\left(x^{d_q} + \cdots\right)^{d_p}$, which will have a leading term $x^{d_p d_q}$.

   In particular, notice that by plugging in $Q$ into $P$, we're essentially raising $Q(x)$ to the power of $1, 2, \ldots, d_p$, and adding them all together (with some potential constants, but we're only interested in the exponents). The largest term that arises when doing this comes from raising $x^{d_q}$ to the power of $d_p$, which gives us a leading term of $x^{d_p d_q}$ for the polynomial $P(Q(x))$ with degree $d_p d_q$.

   (c) What is the maximum degree of $P(xQ(x))$?

   $\boxed{d_p(d_q + 1)}$

   **Answer:** Similar to the previous part, here we're multiplying $Q(x)$ by $x$ before plugging it into $P$. With respect to the exponents, what we're essentially doing is increase the power of all the terms in $Q$ by 1, and then plugging it into $P$.

   With the same reasoning, the leading term of $xQ(x)$ would have exponent $d_q + 1$, and after raising this leading term to the power of $d_p$, we have a leading term of $x^{d_p(d_q+1)}$, with degree $d_p(d_q + 1)$.

2. If a polynomial $P(x)$ has a root at $r$, then $P(x) = (x - r)Q(x)$ for some polynomial $Q(x)$.

   ● **True** ○ False

   **Answer:** We can use long division to get this factorization.

   In particular, long division guarantees that dividing $P(x)$ by $(x - r)$ will give us an expression $P(x) = (x - r)Q(x) + R(x)$ for some polynomial $Q(x)$ and rational function $R(x)$. Further, $R(x)$ must have a degree strictly less than $(x-r)$, i.e. of degree zero and is a constant. Since we know that $P(r)=0$, we can plug $x=r$ on the RHS, giving us $(r-r)Q(r)+R(r)=R(r)=0$.

This suggests that $R(x)=0$ and we can simplify $P(x)=(x-r)Q(x)$ for some polynomial $Q(x)$.

3. Given a degree $d$ polynomial $P(x)$ and $k$ values $x_1,\ldots,x_k$ with $k \leq d$, how many polynomials of degree at most $d$ over arithmetic modulo prime $p$ have the same value as $P9x)$ on $x_1,\ldots,x_k$?

$$\boxed{p^{d+1-k}}$$

**Answer:** Note that any polynomial of degree $d$ is uniquely defined by $d+1$ points. Since we're fixing $k$ of these points, we have $d+1-k$ points left to choose for this new polynomial.

Each one of these points has $p$ possible values, so we have a total of $p^{d+1-k}$ possibilities for choosing the other $d+1-k$ points, each of which defines a unique polynomial.

4. For this problem, consider the Berlekamp–Welch scheme for a message of size $n$ that tolerates $k$ errors.

   (a) What is the degree of the polynomial $P(x)$ used to encode the message?

   $$\boxed{n-1}$$

   **Answer:** Since we want to encode $n$ points into the polynomial, the polynomial will have degree (at most) $n-1$.

   (b) What is the degree of the error locator polynomial $E(x)$ in the reconstruction algorithm?

   $$\boxed{k}$$

   **Answer:** We want to tolerate against $k$ errors, and the error locator polynomial has exactly $k$ factors of $(x-e_i)$, each corresponding to the location $e_i$ of an error in the received message. This means that the error locator polynomial has degree $k$.

   (c) What is the degree of $Q(x)=P(x)E(x)$ in the reconstruction algorithm?

   $$\boxed{n+k-1}$$

   **Answer:** We can just multiply $P(x)E(x)$; the degree of this new polynomial comes from its leading term, which is the product of the leading terms of $P(x)$ and $E(x)$. Since these leading terms have exponents $n-1$ and $k$ respectively, the degree of $Q(x)$ is then $n+k-1$.

   (d) If there were $i \leq k$ errors, the recovered $E(x)$ has at least ________________ roots. Give the largest lower bound you can.

$\boxed{i}$

**Answer:** The error locator polynomial must be equal to zero at these $i$ error locations, and we don't care about where the other zeroes go; this means that at minimum we must have $i$ roots, one for each error location, but we could have more.

## 10 Polynomials and Functions

1. Consider any function $f(x)$ where the domain and range is arithmetic modulo a prime $p$.

   (a) Prove that $f(x)$ corresponds to a polynomial expression modulo $p$.

   **Answer:** Since the domain of $f$ is arithmetic modulo $p$, $f$ is defined on $p$ different $x$-values, so we have a total of $p$ points that define $f(x)$. Using Lagrange interpolation, we can use these $p$ points to construct a degree $p-1$ polynomial, which agrees with $f(x)$ on all of its points mod $p$.

   (b) Give a tight upper bound on the minimum degree of a polynomial that represents $f(x)$. (Your proof above may be useful.)

   $\boxed{p-1}$

   **Answer:** With the same reasoning as above, we have $p$ points that define $f$, and any $p$ points uniquely define a degree at most $p-1$ polynomial.

2. Consider a composite number $m = p_1p_2$, with $p_1$ and $p_2$ different primes, and a polynomial $P(x)$ over arithmetic modulo $m$.

   (a) Give a tight upper bound on the minimum degree of a polynomial expression equivalent as functions to $P(x)$ (i.e. gives the same value when evaluated at all $x$). (Hint: Think about question 1(b) and CRT.)

   $\boxed{\max(p_1-1, p_2-1)}$

   (b) Justify your answer above. That is, show that any polynomial over arithmetic modulo $m$ can be represented by an expression of degree at most your answer above.

   **Answer:** Consider a polynomial $P(x)$ over arithmetic modulo $m$; suppose we define $P_1(x) \equiv P(x) \pmod{p_1}$ and $P_2(x) \equiv P(x) \pmod{p_2}$.

   We know that $P_1$ has degree at most $p_1 - 1$, and $P_2$ has degree at most $p_2 - 1$ by FLT (as any term $x^k$ for $k \geq p$ can be simplified and reduced).

   With this in mind, we can combine $P_1$ and $P_2$ to construct a new polynomial $Q(x)$ such that

   $$Q(x) \equiv P_1(x) \pmod{p_1}$$
   $$Q(x) \equiv P_2(x) \pmod{p_2}$$

   In particular, for any fixed $x$, CRT guarantees a unique solution for $Q(x) \pmod{p_1p_2}$, matching with the value of $P(x)$ exactly at each location (as we've defined $P_1(x)$ and $P_2(x)$ to match exactly with $P(x)$ under mod $p_1$ and $p_2$ respectively).

To find out what $Q(x)$ actually is, we can use the formula for CRT to arrive at

$$Q(x) \equiv P_1(x) \cdot p_2 \cdot \left(p_2^{-1} \bmod p_1\right) + P_2(x) \cdot p_1 \cdot \left(p_1^{-1} \bmod p_2\right) \pmod{p_1 p_2}.$$

Since $p_2 \cdot \left(p_2^{-1} \bmod p_1\right)$ and $p_1 \cdot \left(p_1^{-1} \bmod p_2\right)$ are both just constants, the degree of $Q(x)$ is equal to the larger of the degrees of $P_1(x)$ and $P_2(x)$, i.e. $\max(p_1 - 1, p_2 - 1)$.

(c) Any function under arithmetic modulo $m$ corresponds to a polynomial.

○ True ● **False**

**Answer:** There aren't enough polynomials of degree $\max(p_1 - 1, p_2 - 1)$ to represent all functions under modulo $m$.

Specifically, there are a total of $m^m$ different functions under modulo $m$ (i.e. each of the $m$ possible $x$-values has a total of $m$ possible $y$-values), but only $m^{\max(p_1,p_2)}$ possible polynomials, since the maximum degree is $\max(p_1 - 1, p_2 - 1)$, so we have $\max(p_1, p_2)$ points to define the polynomial with (i.e. one more than the degree; each of the $\max(p_1, p_2)$ points has $m$ possibilities for $y$-values).

Since $m^m > m^{\max(p_1,p_2)}$, there are more possible functions than possible polynomials, so there are functions that do not correspond to any polynomial under modulo $m$.

## 11 Count the Ways

Jonathan is playing a game called 70rdle. In this game, the 70 staff has a secret string of five upper-case English letters, and Jonathan must guess exactly what the string is. Assume each subpart is independent of the other subparts. The English alphabet has 26 letters.

1. With no restrictions, how many strings are possible?

$$\boxed{26^5}$$

**Answer:** We have 26 choices for the first letter, 26 choices for the second letter, etc. This means that the total number of strings is $26 \cdot 26 \cdot 26 \cdot 26 \cdot 26 = 26^5$.

2. If no letter is allowed to appear in the string more than once, how many strings are possible?

$$\boxed{\frac{26!}{21!}}$$

**Answer:** Since all letters must be distinct, we can first choose the set of 5 letters out of the 26 total letters to include in the string; there are $\binom{26}{5}$ ways to do this. Next, we can order these 5 distinct letters in any way; there are 5! ways to do this. This makes the final answer

$$\binom{26}{5} 5! = \frac{26!}{21!5!} 5! = \frac{26!}{21!}.$$

3. If the letters in the string must be sorted in alphabetical order, how many strings are possible? For example, AABCD is valid, but ABCDA is not.

$$\binom{30}{5} = \binom{30}{25}$$

**Answer:** We can view this as a stars and bars problem—order doesn't matter (as we will always have exactly one ordering of the letters we pick), but we choose with replacement (as we can have duplicate letters).

There are 5 stars (the letters we choose), and 26 categories (one for each kind of letter), meaning we have 25 bars. This gives an answer of $\binom{25+5}{5} = \binom{30}{5}$ or $\binom{25+5}{25} = \binom{30}{25}$.

4. How many strings contain at least one J?

$$26^5 - 25^5$$

**Answer:** It's perhaps easier to count the strings that contain no J's (i.e. the complement of what we're actually trying to count). In order to choose a string with no J's, we can just remove J from consideration, leaving only 25 letters to choose from. This gives us $25^5$ possible strings with no J's.

Subtracting from the total of $26^5$ possible strings we can make, the total number of strings with at least one J is $26^5 - 25^5$.

5. How many strings contain at least one J in the first two letters?

$$2 \cdot 26^4 - 26^3$$

**Answer:** We can use the principle of inclusion-exclusion here. There are a total of $26^4$ strings with a J as the first letter, and $26^4$ strings with a J as the second letter. However, we've overcounted—the strings with J as *both* the first and second letter are counted twice. This means that we need to subtract this overcounting—there are $26^3$ such strings with J as both the first and second letter.

This gives an answer of $2 \cdot 26^4 - 26^3$.

6. How many strings contain exactly four J's?

125

**Answer:** A string with exactly four J's has only one letter that is not a J; there are 25 choices for what this other letter is. We can also put this letter anywhere we want, in any of the 5 positions in the string, so in total we have $5 \cdot 25 = 125$ possible strings.

7. How many strings contain exactly five J's?

1

**Answer:** There is only one string: JJJJJ.

## 12 A little bit of Fermat from Induction

Suppose $p$ is a prime.

1. Prove that for any $1 \le k \le p-1$, $p \mid \binom{p}{k}$.

   **Answer:** We may express

   $$\binom{p}{k} = \frac{p!}{k!(p-k)!}.$$

   Since $1 \le k \le p-1$, neither $k!$ nor $(p-k)!$ have factors of $p$, but $p!$ has a factor of $p$.

   This means that the factor of $p$ in the numerator is not canceled out by any factors of $p$ in the denominator, and $p \mid \binom{p}{k}$.

2. Prove Fermat's Little Theorem via induction: in other words, prove that for all integers $a$,

   $$a^p \equiv a \pmod{p}.$$

   (Hint: You may find the Binomial Theorem helpful: $(x+y)^p = \sum_{i=0}^{p} \binom{p}{i} x^{p-i} y^i$.)

   **Answer:** We proceed by induction on $a$.

   **Base case** ($a = 0$): We have $a^p \equiv 0^p \equiv 0 \equiv a \pmod{p}$.

   **Induction Hypothesis**: Suppose $a^p \equiv a \pmod{p}$.

   **Inductive Step**: We will show the case for $a+1$, i.e. we will show that $(a+1)^p \equiv a+1 \pmod{p}$.

   Applying the Binomial Theorem, we have

   $$\begin{aligned}(a+1)^p &= \sum_{i=0}^{p} \binom{p}{i} a^{p-i} 1^i \\ &= \sum_{i=0}^{p} \binom{p}{i} a^{p-i} \\ &= \binom{p}{0} a^p + \binom{p}{1} a^{p-1} + \binom{p}{2} a^{p-2} + \cdots + \binom{p}{p-1} a + \binom{p}{p}\end{aligned}$$

   Notice that from part (a), any $\binom{p}{k}$ for $1 \le k \le p-1$ are all multiples of $p$; this means that $\binom{p}{k} \equiv 0 \pmod{p}$ for all $1 \le k \le p-1$. If we take the above expansion modulo $p$, all the terms cancel out except for the terms with $\binom{p}{0}$ and $\binom{p}{p}$:

   $$\begin{aligned}&\equiv \binom{p}{0} a^p + \binom{p}{p} \pmod{p} \\ &= a^p + 1 \pmod{p} \\ &\equiv a + 1 \pmod{p}\end{aligned}$$

   Here, in the last equivalence we used the inductive hypothesis to simplify $a^p \pmod{p}$. This proves the claim for $a+1$, and by the principles of induction, the claim holds for all $a \ge 0$.

   We also want to show the claim for negative $a$; this follows from the equivalence of all integers into a residue class in $\{0, \dots, p-1\}$. In particular, all negative integers are equivalent to some number in $\{0, \dots, p-1\}$, and we've proven such cases through induction.

   This means that the claim holds for all integers $a$, as desired.

# 13 Rare Scheme for Alice

Bob is trying to send a message $m$ to Alice$_1$, ..., Alice$_M$, such that if $k$ of the Alices agree, they can construct the message. However, he wants to prevent Eve from reconstructing the message, even if she is able to determine all of the Alices' points.

The Alices use the same RSA scheme, broadcasting the public key $(N, e)$. Bob then uses the polynomial secret sharing scheme (over arithmetic modulo a sufficiently large prime $p$), sending points $(1, x_1), \dots, (M, x_M)$ to Alice$_1$, ..., Alice$_M$, respectively, such that if $P(x)$ is the polynomial determined by the points, then $P(0)$ is $m^e$.

1. After completing the interpolation, how can the Alices reconstruct $m$?

   **Answer:** Since the Alices know $d$, they can compute $(m^e)^d \equiv m^{ed} \equiv m \pmod{N}$; the correctness follows from RSA.

2. Eve doesn't know the Alices' private key $d$, but knows $N$, $e$, and $p$. She manages to trick all of the Alices into revealing $(1, x_1^d), \dots, (M, x_M^d)$. Prove that if $M > (k-1)d$, then Eve can find the message $m$ without figuring out the value of $d$.

   **Answer:** Eve can perform Lagrange interpolation on the $(i, x_i^d)$ points to get a unique polynomial $Q(x)$ of degree at most $M-1$. We claim that $Q(x) = P(x)^d$.

   Note that for $1 \le i \le m$, we have $Q(i) = x_i^d = P(i)^d$. If we consider the polynomials $Q(x)$ and $P(x)^d$, we can see that both polynomials agree on these $M$ points.

   Further, we know that $P(x)$ has degree $k-1$, since we want the $k$ Alices to be able to reconstruct $P(x)$ with their $k$ points. This means that $P(x)^d$ has degree $d(k-1)$.

   Looking at the difference of the two polynomials, we can see that $Q(x) - P(x)^d$ has degree $\max(M-1, d(k-1)) \le M-1$, since $M > d(k-1)$. Because $Q(x) - P(x)^d = 0$ at the $M$ points $x_1, \dots, x_M$, and it has degree at most $M-1$, it is uniquely defined by these $M$ points, and must be the zero polynomial.

   This means that $Q(x) - P(x)^d = 0$, which implies that $Q(x) = P(x)^d$. In particular, to get the secret message, we know that $Q(0) = P(0)^d = (m^e)^d = m^{ed} = m$.

   Throughout this process, Eve has no knowledge of $d$, only $P(x)^d$, which gives no information about $d$; however, Eve was still able to reconstruct the message $m$.