CS 70 Discrete Mathematics and Probability Theory
Spring 2016 Rao and Walrand HW 1

Due Thursday January 28 at 10PM

1. (3 points) Wason's experiment:2

   Suppose we have four cards on a table:

   - 1st about Alice, 2nd about Bob, 3rd about Charlie, and 4th about Donna.
   - For each person, one side of the card indicates their dessert, the other what they did after dinner.
   - Theory: "If a person has ice cream for dessert, he/she has to do the dishes after dinner."
   - Cards: Alice: fruit, Bob: watched TV, Charlie: ice cream, Donna: did dishes

   Whose cards do you have to flip to test the theory? **Answer:** **Answer:** Charlie's and Bob's.

   From the theory we know that "eating ice cream" implies "doing the dishes", and we know the contropositive is true as well: "not doing dishes" implies "not eating ice cream".

   Therefore, we need to check if Charlie actually did the dishes, and we need to make sure Bob did not eat ice cream.

2. (8 points) Prove or Disprove.

   (a) $A \vee (B \wedge C) \equiv (A \vee B) \wedge (A \vee C)$.

   Answer: **Answer**: True.

A	B	C	$A \vee (B \wedge C)$	$(A \vee B) \wedge (A \vee C)$
T	T	T	T	T
T	T	F	T	T
T	F	T	T	T
T	F	F	T	T
F	T	T	T	T
F	T	F	F	F
F	F	T	F	F
F	F	F	F	F

   (b) $A \wedge (B \vee C) \equiv (A \wedge B) \vee (A \wedge C)$.

   Answer: **Answer**: True.

A	B	C	$A \wedge (B \vee C)$	$(A \wedge B) \vee (A \wedge C)$
T	T	T	T	T
T	T	F	T	T
T	F	T	T	T
T	F	F	F	F
F	T	T	F	F
F	T	F	F	F
F	F	T	F	F
F	F	F	F	F

(c) $A \implies (B \wedge C) \equiv (A \implies B) \wedge (A \implies C)$

Answer: **Answer**: True.

A	B	C	$A \implies (B \wedge C)$	$(A \implies B) \wedge (A \implies C)$
T	T	T	T	T
T	T	F	F	F
T	F	T	F	F
T	F	F	F	F
F	T	T	T	T
F	T	F	T	T
F	F	T	T	T
F	F	F	T	T

(d) $A \implies (B \vee C) \equiv (A \implies B) \vee (A \implies C)$

Answer: **Answer**: True.

A	B	C	$A \implies (B \vee C)$	$(A \implies B) \vee (A \implies C)$
T	T	T	T	T
T	T	F	T	T
T	F	T	T	T
T	F	F	F	F
F	T	T	T	T
F	T	F	T	T
F	F	T	T	T
F	F	F	T	T

3. (9 points) Determine whether the following equivalences hold, and give brief justifications for your answers. Clearly state whether or not each pair is equivalent.

   (a) (3 points) $\forall x\, \exists y\, \big(P(x) \Rightarrow Q(x,y)\big) \equiv \forall x\, \big(P(x) \Rightarrow (\exists y\, Q(x,y))\big)$

   Answer: **Claim**: $\forall x\, \exists y\, \big(P(x) \Rightarrow Q(x,y)\big) \equiv \forall x\, \big(P(x) \Rightarrow (\exists y\, Q(x,y))\big)$
   Answer: The equivalence holds.
   Justification: We can rewrite the claim as $\forall x\, \exists y\, \big(\neg P(x) \vee Q(x,y)\big) \equiv \forall x\, \big(\neg P(x) \vee (\exists y\, Q(x,y))\big)$. Clearly, the two sides are the same if $\neg P(x)$ is true. If $\neg P(x)$ is false, then the two sides are still the same, because $\forall x\, \exists y\, \big(\text{False} \vee Q(x,y)\big) \equiv \forall x\, \big(\text{False} \vee (\exists y\, Q(x,y))\big)$.

   (b) (3 points) $\neg\exists x\, \forall y\, \big(P(x) \Rightarrow \neg Q(x,y)\big) \equiv \forall x\, \exists y\, \big(P(x) \wedge Q(x,y)\big)$

   Answer: **Claim**: $\neg\exists x\, \forall y\, \big(P(x) \Rightarrow \neg Q(x,y)\big) \equiv \forall x\, \exists y\, \big(P(x) \wedge Q(x,y)\big)$ **Answer**: The equivalence holds.
   Justification: Truth tables show that $P(x) \Rightarrow \neg Q(x,y) \equiv \neg P(x) \vee \neg Q(x,y)$. Using De Morgan's Law to distribute the negation on the left side yields $\forall x\, \exists y\, \big(\neg\neg P(x) \wedge \neg\neg Q(x,y)\big)$, which is equivalent to the right side.

   (c) (3 points) $\forall x\, \exists y\, \big(Q(x,y) \Rightarrow P(x)\big) \equiv \forall x\, \big((\exists y\, Q(x,y)) \Rightarrow P(x)\big)$

   Answer: **Claim**: $\forall x\, \big((\exists y\, Q(x,y)) \Rightarrow P(x)\big) \equiv \forall x\, \exists y\, \big(Q(x,y) \Rightarrow P(x)\big)$ **Answer**: The equivalence does not hold.
   Justification: We can rewrite the claim as $\forall x\, \big((\neg(\exists y\, Q(x,y))) \vee P(x)\big) \equiv \forall x\, \exists y\, \big(\neg Q(x,y) \vee P(x)\big)$ By De Morgan's Law, distributing the negation on the right side of the equivalence changes the $\exists y$ to $\forall y$, and the two sides are clearly not the same. Another approach to the problem is to consider by linguistic example. Let x and y span the universe of all people, and let $Q(x,y)$ mean "Person x is Person y's offspring", and let $P(x)$ mean "Person x likes tofu". The

right side claims that, for all Persons x, there exists some Person y such that either Person x is not Person y's offspring or that Person x likes tofu. The left side claims that, for all Persons x, if there exists a parent of Person x, then Person x likes tofu. Obviously, these are not the same.

4. (9 points) Decide whether each of the following propositions is true, when the domain for x and y is the real numbers $\mathbb{R}$. Prove your answers.

   (a) (3 points) $\forall x \exists y \ (xy > 0 \Rightarrow y > 0)$

   Answer: **Answer**: True.
   Proof: The antecedant is false $y = 0$.
   Because of this, the implication is vacuously true. ■

   (b) (3 points) $\neg \forall x \exists y \ (xy \geq x^2)$ **Answer:** False.

   Look at the proposition before the negation.

   Claim: $\forall x \exists y \ (xy \geq x^2)$
   Answer: True.
   Proof: Let $y = x$. It is trivially true that $\forall x \ (x^2 \geq x^2)$. ■ Thus, the negation is False.

   (c) (3 points) $\exists y \forall x \ (xy \geq x^2)$ **Answer:** **Claim**: $\exists y \forall x \ (xy \geq x^2)$

   Answer: False.
   Proof: The proposition cannot be true for some $y < 0$, since $x^2 \geq 0$ and $xy < 0$ for $x > 0$ and $y < 0$. The proposition similarly cannot be true for some $y > 0$, since $x^2 \geq 0$ and $xy < 0$ for $x < 0$ and $y < 0$. The proposition is obviously not true for $y = 0$, since $x^2 > 0$ for $x \neq 0$. Since the proposition cannot be true for any real number y, the proposition is false. ■

5. (7 points) Here are statements about a magical world:

   (I) Duck Dynasty viewers don't read the candidates' positions.

   (II) No one, who votes, ever fails to do their homework (on the issues).

   (III) No one is well-informed, if he or she is confused.

   (IV) Everyone who has done their homework (on the issues) is well-informed.

   (V) A person is always confused if he or she doesn't read the candidates positions.

   (VI) No one wears a party hat, unless he or she votes.

   (a) (3 points) Write each of the above six sentences as a quantified proposition over the universe of all people. You should use the following symbols for the various elementary propositions: $V(x)$ for "x votes", $H(x)$ for "x has done their homework", $W(x)$ for "x is well-informed", $C(x)$ for "x is confused", $D(x)$ for "x is a Duck Dynasty viewer", $I(x)$ for "x doesn't read the candidates' positions", and $P(x)$ for "x wears a party hat".

   Answer:

   (I) **Answer**: $\forall x \left(D(x) \Rightarrow I(x)\right)$
   (II) **Answer**: $\forall x \left(V(x) \Rightarrow H(x)\right)$
   (III) **Answer**: $\forall x \left(C(x) \Rightarrow \neg W(x)\right)$
   (IV) **Answer**: $\forall x \left(H(x) \Rightarrow W(x)\right)$
   (V) **Answer**: $\forall x \left(I(x) \Rightarrow C(x)\right)$
   (VI) **Answer**: $\forall x \left(P(x) \Rightarrow V(x)\right)$

   (b) (2 points) Now rewrite each proposition equivalently using the contrapositive.

   Answer:

   (I) $\forall x \left(\neg I(x) \Rightarrow \neg D(x)\right)$
   (II) $\forall x \left(\neg H(x) \Rightarrow \neg V(x)\right)$
   (III) $\forall x \left(W(x) \Rightarrow \neg C(x)\right)$
   (IV) $\forall x \left(\neg W(x) \Rightarrow \neg H(x)\right)$
   (V) $\forall x \left(\neg C(x) \Rightarrow \neg I(x)\right)$
   (VI) $\forall x \left(\neg V(x) \Rightarrow \neg P(x)\right)$

   (c) (2 points) You now have twelve propositions in total. What can you conclude from them about a person who wears a party hat? Explain clearly the implications you used to arrive at your conclusion.

   Answer: **Answer**: A person who wears a party hat is not an Duck Dynasty watcher.
   Derivation: $P(x) \Rightarrow V(x) \Rightarrow H(x) \Rightarrow W(x) \Rightarrow \neg C(x) \Rightarrow \neg I(x) \Rightarrow \neg D(x)$

6. (20 points) Karnaugh Maps

   Below is the truth table where F is encoded as 0 and T is encoded as 1 for the boolean function

   $$Y = (\neg A \wedge \neg B \wedge C) \vee (\neg A \wedge B \wedge \neg C) \vee (A \wedge \neg B \wedge C) \vee (A \wedge B \wedge C).$$

A	B	C	Y
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

In this question, we will explore a different way of representing a truth table, the *Karnaugh map*. A Karnaugh map is just a grid-like representation of a truth table, but as we will see, the mode of presentation can give more insight. The values inside the squares are copied from the output column of the truth table, so there is one square in the map for every row in the truth table.

Around the edge of the Karnaugh map are the values of the input variables, where again F is encoded as 0 and T is encoded is 1. Note that the sequence of numbers across the top of the map is not in binary sequence, which would be 00, 01, 10, 11. It is instead 00, 01, 11, 10, which is called *Gray code* sequence. Gray code sequence only changes one binary bit as we go from one number to the next in the sequence. That means that adjacent cells will only vary by one bit, or Boolean variable. In other words, *cells sharing common Boolean variable values are adjacent.*

For example, here is the Karnaugh map for Y:

		BC			
		00	01	11	10
A	0	0	1	0	1
	1	0	1	1	0

The Karnaugh map provides a simple and straight-forward method of minimizing boolean expressions by visual inspection. The technique is to examine the Karnaugh map for any groups of adjacent ones that occur, which can be combined to simplify the expression. Note that "adjacent" here means in the modular sense, so adjacency wraps around the top/bottom and left/right of the Karnaugh map; for example, the top-most cell of a column is adjacent to the bottom-most cell of the column.

For example, the ones in the second column in the Karnaugh map above can be combined because $(\neg A \wedge \neg B \wedge C) \vee (A \wedge \neg B \wedge C)$ simplifies to $(\neg B \wedge C)$. Applying this technique to the Karnaugh map (illustrated below), we obtain the following simplified expression for Y:

$$Y = (\neg B \wedge C) \vee (A \wedge C) \vee (\neg A \wedge B \wedge \neg C).$$

		BC			
		00	01	11	10
A	0	0	1	0	1
	1	0	1	1	0

Answer:

(a) Write the truth table for the boolean function

$$Z = (\neg A \wedge \neg B \wedge \neg C \wedge \neg D) \vee (\neg A \wedge \neg B \wedge C \wedge \neg D) \vee (A \wedge \neg B \wedge \neg C \wedge \neg D) \vee (A \wedge \neg B \wedge C \wedge \neg D).$$

A	B	C	D	Z
0	0	0	0	1
0	0	0	1	0
0	0	1	0	1
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	1
1	0	0	1	0
1	0	1	0	1
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

(b) Using your truth table from Part 1, fill in the Karnaugh map for Z below.

AB \ CD	00	01	11	10
00				
01				
11				
10				

AB \ CD	00	01	11	10
00	1	0	0	1
01	0	0	0	0
11	0	0	0	0
10	1	0	0	1

(c) Using your Karnaugh map from Part 2, write down a simplified expression for Z.

The four corners can be combined to get

$$Z = \neg B \wedge \neg D.$$

The entire map can be wrapped onto a torus (a donut shape - the way that video games [like Pac-Man or Asteroids] sometimes wrap around so if you move off the right side, you come out the left side, and if you move past the top, you come out the bottom). The ones form a square with only B and D remaining unchanged at 0 and 0 whereas A and C takes on the values $(00, 01, 10, 11)$ which constitutes all possible combinations AC can take.

AB \ CD	00	01	11	10
00	1	0	0	1
01	0	0	0	0
11	0	0	0	0
10	1	0	0	1

(d) Show that this simplification could also be found algebraically by factoring the expression for Z in (1).

$$Z = (\neg A \wedge \neg B \wedge \neg C \wedge \neg D) \vee (\neg A \wedge \neg B \wedge C \wedge \neg D) \vee (A \wedge \neg B \wedge \neg C \wedge \neg D) \vee (A \wedge \neg B \wedge C \wedge \neg D)$$

By using the distributive law $(A \wedge B) \vee (A \wedge C) = A \wedge (B \vee C)$, we get the following.

$$Z = (\neg B \wedge \neg D) \wedge ((\neg A \wedge \neg C) \vee (\neg A \wedge C) \vee (A \wedge \neg C) \vee (A \wedge C))$$

As $((\neg A \wedge \neg C) \vee (\neg A \wedge C) \vee (A \wedge \neg C) \vee (A \wedge C))$ is a tautology (fancy word meaning always true no matter what), we get the following simplification.

$$Z = (\neg B \wedge \neg D) \wedge (1)$$

$$Z = (\neg B \wedge \neg D)$$

7. (5 points) Proof by?

(a) (3 points) Prove that if $x, y, a \in \mathbb{Z}$, if a does not divide xy, then a does not divide x and a does not divide y. In notation: $(\forall x, y \in \mathbb{Z})\ a \nmid xy \implies (a \nmid x \ \wedge \ a \nmid y)$. What proof technique did you use?

Answer: We will use proof by contraposition. For any arbitrary given x and y, the statement $a \nmid xy \implies (a \nmid x \wedge a \nmid y)$ is equivalent using contraposition to $\neg(a \nmid x \wedge a \nmid y) \implies \neg(a \nmid xy)$. Moving the negations inside, this becomes equivalent to $(a \mid x \vee a \mid y) \implies a \mid xy$.

Now for this part, we give a proof by cases. Assuming that $a \mid x \vee a \mid y$, one of the two cases must be true.

i. $a \mid x$: in this case $x = ak$ for some $k \in \mathbb{Z}$. Therefore $xy = aky$ which is a multiple of a. So $a \mid xy$.

ii. $a \mid y$: in this case $y = ak$ for some $k \in \mathbb{Z}$. Therefore $xy = akx$ which is a multiple of a. So $a \mid xy$.

Therefore assuming $a \mid x \vee a \mid y$ we proved $a \mid xy$.

We used proof by cases and proof by contraposition.

(b) (1 point) Prove or disprove the contrapositive.

Answer: We proved the statement. The contrapositive of a statement has logically equivalent to the statement. So we are done.

(c) (1 point) Prove or disprove the converse.

Answer: Its not true! The converse is that if a does not divide x and does not divide y than a does not divide xy. We can choose $x = 2$ and $y = 5$ and see a counterexample to the statement.

8. (18 points) Prove or disprove each of the following statements. For each proof, state which of the proof types (as discussed in Note 2) you used.

(a) (3 points) For all natural numbers n, if n is odd then $n^2 + 3n$ is even. **Answer:** **Claim**: For all natural numbers n, if n is odd then $n^2 + 3n$ is even.
Answer: True.
Proof: We will use a direct proof. Assume n is odd. By the definition of odd numbers, $n = 2k+1$ for some natural number k. Substituting into the expression n^2+3n, we get $(2k+1)^2+3\times(2k+1)$. Simplifying the expression yields $4k^2+10k+4$. This can be rewritten as $2\times(2k^2+5k+2)$. Since $2k^2+5k+2$ is a natural number, by the definition of even numbers, n^2+3n is even. ■

(b) (3 points) For all natural numbers n, $n^2 + 7n$ is even.

Answer: **Claim**: For all natural numbers n, n^2+7n is even.
Answer: True.
Proof: We will use a proof by cases. Let n be an even number. By the definition of even numbers, $n = 2k$ for some natural number k. Substituting into the expression n^2+7n, we get $(2k)^2+7\times(2k)$. Simplifying the expression yields $4k^2+14k$. This can be rewritten as $2\times(2k^2+7k)$, which is an even number. Therefore, if n is even, then n^2+7n is even. Now let n be an odd number. By the definition of odd numbers, $n = 2k+1$ for some natural number k. Substituting into the expression n^2+7n, we get $(2k+1)^2+7\times(2k+1)$. Simplifying the expression yields $4k^2+18k+8$. This can be rewritten as $2\times(2k^2+9k+4)$, which is an even number. Therefore, if n is odd, then n^2+7n is even. Since n^2+7n is even when n is even or when n is odd, n^2+7n is even for all natural numbers n. ■

(c) (3 points) For all real numbers a, b, if $a+b \geq 10$ then $a \geq 7$ or $b \geq 3$.

Answer: **Claim**: For all real numbers a, b, if $a+b \geq 10$ then $a \geq 7$ or $b \geq 3$.
Answer: True.
Proof: We will use a proof by contraposition. Suppose that $a < 7$ and $b < 3$ (note that this is equivalent to $\neg\big(a \geq 7 \vee b \geq 3\big)$). Since $a < 7$ and $b < 3$, $a+b < 10$ (note that $a+b < 10$ is equivalent to $\neg\big(a+b \geq 10\big)$). Thus, if $a+b \geq 10$, then $a \geq 7$ or $b \geq 3$ (or both, as "or" is not "exclusive or" in this case). ■

(d) (3 points) For all real numbers r, if r is irrational then $r+1$ is irrational.

Answer: **Claim**: For all real numbers r, if r is irrational then $r-1$ is irrational.
Answer: True.

Proof: We will use a proof by contraposition. Assume that $r+1$ is rational. Since $r-1$ is rational, it can be written in the form a/b where a and b are integers. Then r can be written as $(a+b)/b$. By the definition of rational numbers, r is a rational number, since both $a+b$ and b are integers. By contraposition, if r is irrational, then $r+1$ is irrational. ■

(e) (3 points) For all natural numbers n, $10n^2 > n!$.

Answer: Claim: For all natural numbers n, $10n^2 > n!$.
Answer: False.
Proof: We will use proof by counterexample. Let $n = 6$. $10 \times 6^2 = 360$. $6! = 720$. Since $10n^2 < n!$, the claim is false. ■

(f) (3 points) For all natural numbers a where a^5 is odd, then a is odd. **Answer: Claim: For all natural numbers if a^5 is odd, then a is odd.**
Answer: True.
Proof: This will be proof by contrapositive. The contrapositive is "If a is even, then a^5 is even." Let a be even. By the definition of even, $a = 2k$. Then $a^5 = (2k)^5 = 2(16k^5)$, which implies a^5 even. ■

CS 70 Discrete Mathematics and Probability Theory
Spring 2016 Rao and Walrand HW 2

Due Thursday February 4th at 10PM

1. (5 points)

   Use induction to prove that for all positive integers n, all of the entries in the matrix

   $$\begin{pmatrix} 1 & 0 \\ 3 & 1 \end{pmatrix}^n$$

   are $\leq 3n$.

   Answer:

   Before starting the proof, writing out the first few powers reveals a telling pattern:

   $$\begin{pmatrix} 1 & 0 \\ 3 & 1 \end{pmatrix}^1 = \begin{pmatrix} 1 & 0 \\ 3 & 1 \end{pmatrix}$$
   $$\begin{pmatrix} 1 & 0 \\ 3 & 1 \end{pmatrix}^2 = \begin{pmatrix} 1 & 0 \\ 6 & 1 \end{pmatrix}$$
   $$\begin{pmatrix} 1 & 0 \\ 3 & 1 \end{pmatrix}^3 = \begin{pmatrix} 1 & 0 \\ 9 & 1 \end{pmatrix}$$

   It appears (and we shall soon prove) that the upper left and lower right entries are always 1, the upper right entry is always 0, and the lower left entry is $3n$. We shall take this to by our inductive hypothesis. **Proof**: We shall use a proof by induction that the upper left and lower right entries of the matrix are always 1, the upper right entry is always 0, and the lower left entry is $3n$. This will prove that all entries in the matrix are less than or equal to $3n$ for all $n \geq 1$. The base case of $n = 1$ is trivially true. Now suppose that our proposition is true for some $n \geq 1$, meaning

   $$\begin{pmatrix} 1 & 0 \\ 3 & 1 \end{pmatrix}^n = \begin{pmatrix} 1 & 0 \\ 3n & 1 \end{pmatrix}$$

   for some $n \geq 1$. Multiplying both sides of the equation by the original matrix yields

   $$\begin{pmatrix} 1 & 0 \\ 3 & 1 \end{pmatrix}^{n+1} = \begin{pmatrix} 1 & 0 \\ 3n & 1 \end{pmatrix}\begin{pmatrix} 1 & 0 \\ 3 & 1 \end{pmatrix} = \begin{pmatrix} 1+0 & 0 \\ 3n+3 & 0+1 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 3(n+1) & 1 \end{pmatrix}$$

   By the principle of induction, our proposition is therefore true for all $n \geq 1$, so all entries in the matrix will be less than or equal to $3n$. ■

2. (5 points) Divergence of harmonic series

   You may have seen the series $1 + \frac{1}{2} + \frac{1}{3} + \ldots$ in calculus. This is known as a *harmonic series*, and it diverges, i.e. the sum approaches infinity. We are going to prove this fact using induction.

Let $H_j = \sum_{k=1}^{j} \frac{1}{k}$. Use mathematical induction to show that, for all integers $n \geq 0$, $H_{2^n} \geq 1 + \frac{n}{2}$, thus showing that H_j must grow unboundedly as $j \to \infty$. **Answer:**

Base case: $H_{2^0} = H_1 = 1 \geq 1 + \frac{0}{2}$

Inductive Step: Assume that $H_{2^n} \geq 1 + \frac{n}{2}$. Then:

$$\begin{aligned} H_{2^{k+1}} &= 1 + \frac{1}{2} + \ldots + \frac{1}{2^{k+1}} \\ &= (1 + \frac{1}{2} + \ldots + \frac{1}{2^k}) + (\frac{1}{2^k+1} + \ldots + \frac{1}{2^{k+1}}) \\ &= H_{2^k} + (\frac{1}{2^k+1} + \ldots + \frac{1}{2^{k+1}}) \end{aligned}$$

By noting that $(\frac{1}{2^k+1} + \ldots + \frac{1}{2^{k+1}})$ has 2^k terms, each of which is at least $\frac{1}{2^{k+1}}$

$$\geq H_{2^k} + 2^k * \frac{1}{2^{k+1}}$$

By the inductive hypothesis:

$$\begin{aligned} &\geq 1 + \frac{k}{2} + 2^k * \frac{1}{2^{k+1}} \\ &= 1 + \frac{k}{2} + \frac{1}{2} \\ &= 1 + \frac{k+1}{2} \end{aligned}$$

Hence we have proved the statement by induction, and can conclude that H_{2^n} must go to infinity as $n \to \infty$, hence H_n must be diverging as $n \to \infty$.

3. (10 Points)

Prove that for every positive integer k, the following is true:

> For every real number $r > 0$, there are only finitely many solutions in positive integers to $\frac{1}{n_1} + \cdots + \frac{1}{n_k} = r$.

In other words, there exists some number m (that depends on k and r) such that there are at most m ways of choosing a positive integer n_1, and a (possibly different) positive integer n_2, etc., that satisfy the equation.

Hint: You can assume $n_1 \leq n_2 \leq \ldots \leq n_k$ without losing generality. (Why? Think about it)

Answer: **Claim**: $\forall k \in \mathbf{Z}\ \forall r \in \mathbf{R}\ \big((k > 0 \wedge r > 0) \Rightarrow (\text{There are finitely many solutions to } \frac{1}{n_1} + \cdots + \frac{1}{n_k} = r, n_i \in \mathbf{Z}, n_i > 0)\big)$

Proof: We will prove this by induction on k. For our base case, $k = 1$. In the base case, iff r can be written as $\frac{1}{n_1}$ when n_1 is a positive integer, then there is exactly one solution, $n_1 = \frac{1}{r}$. If r cannot be written in that form, then there are exactly zero solutions. In all cases, there is a finite number of solutions. For the inductive hypothesis, assume that there are finitely many solutions for some $k \geq 1$ for all r. Each real number r_1 either can or cannot be written as the sum of $k+1$ integers' inverses. If r_1 cannot be written in that form, then there are exactly zero solutions. If r_1 can be written in that

form, then the integers' inverses can be ordered. Since r_1 is the sum of $k+1$ integers' inverses, the largest $\frac{1}{n_i}$ must be at least $\frac{r_1}{k+1}$. This means that the smallest n_i must be at most $\frac{k+1}{r_1}$, which means that the smallest n_i has finitely many possible values. For each of the possible smallest n_i values, there is a real number $r_1 - \frac{1}{n_i}$ that can be written as the sum of k integers' inverses in finitely many ways (using the induction hypothesis). This means that there are only finitely many possible solutions for $k+1$ (combining all solutions (finitely many) for each possible smallest n_i values (finitely many)). By the principle of induction, there are finitely many solutions for all k for all r.

4. (12 points: 3 each) Objective Preferences

   Imagine that in the context of stable marriage all men have the same preference list. That is to say there is a global ranking of women, and men's preferences are directly determined by that ranking.

   (a) Prove that the first woman in the ranking has to be paired with her first choice in any stable pairing.

   Answer: If the first woman is not paired with her first choice, then she and her first choice would form a rogue couple, because her first choice prefers her over any other woman, and vice versa.

   (b) Prove that the second woman has to be paired with her first choice if that choice is not the same as the first woman's first choice. Otherwise she has to be paired with her second choice.

   Answer: If the first and second women have different first choices, then the second woman must be matched to her first choice. Otherwise she and her first choice would form a rogue couple (since her first choice is not matched to the first woman, he would prefer the second woman over his current match).

   If the first choices are the same, then the second woman must be paired with her second choice, otherwise she and her second choice would form a rogue couple (neither of them are matched to their first choices, and they are each other's second choice).

   (c) Continuing this way, assume that we have determined the pairs for the first $k-1$ women in the ranking. Who should the k-th woman be paired with?

   Answer: The k-th woman should be paired with the first man on her list who has not been matched yet (with the first $k-1$ women). If she's not matched to him, they would form a rogue couple. This is because the man would have to be matched to a woman ranked worse than k, so she would prefer the k-th woman over his current partner, and the k-th woman obviously prefers him to whoever she's matched with.

   (d) Prove that there is a unique stable pairing.

   Answer: In the previous parts, we saw that for each woman, given the pairs for the lower-ranked women, her pair would be determined uniquely. So there is only one stable pairing.

   This can be stated and proved more rigorously using induction. Namely that there is a unique pairing for the first k women, assuming stability. An induction on k would prove this.

5. (15 points:3/3/4/5)

   You have been asked to assign TAs for the fall semester. Each class has its own method for ranking candidates, and each candidate has their own preferences. An assignment is **unstable** if a class and a candidate prefer each other to their current assignments. Otherwise, it is **stable**.

   Candidate information:

Candidate	CS61C Grade	CS70 Grade	CS61A Grade	Teaching Experience	Overall GPA	Preferences
A	A+	A	A	Yes	3.80	CS61C > CS70 > CS61A
B	A	A	A	No	3.61C	CS61C > CS61A > CS70
C	A	A+	A-	Yes	3.60	CS61C > CS70 > CS61A

Ranking method:

- CS61C: Rank by CS61C grade. Break ties using teaching experience, then overall GPA.
- CS70: Rank by teaching experience. Break ties using CS70 grade, then overall GPA.
- CS61A: Rank by CS61A grade. Break ties using overall GPA, then teaching experience.

a) Find a stable assignment.

b) Can you find another, or is there only one stable assignment (if there is only one, why)?

CS61C is overenrolled and needs two TAs. There is another candidate.

Candidate	CS61C Grade	CS70 Grade	CS61A Grade	Teaching Experience	Overall GPA	Preference
D	A+	A	A+	No	3.90	CS70 > CS61A > CS61C

c) Find a stable assignment.

d) Prove your assignment in Part (c) is stable.

Answer:

(a) Use SMA with the following class rankings and students proposing.

CS61C	A > C > B
70	C > A > B
61A	A > B > C

Day	CS61C	CS70	CS61A
1	A, B, C		
2	A	C	B

Assignment: (CS61C, A), (CS70, C), (CS61A, B).

(b) Run SMA with classes proposing and get the same assignment.

Day	A	B	C
1	61C,61A		70
2	61C	61A	70

For any stable assignment, if a student is paired with a class C', the student must prefer his/her optimal class to C' and prefer C' to his/her pessimal class. When students proposed, SMA outputs the student optimal assignment. When classes proposed, SMA outputs the class optimal assignment, which is the student pessimal assignment. Because the student optimal assignment is the same as the student pessimal assignment, C' can only be the class in the assignment for any student, and thus there can only be one stable assignment.

(c) Use SMA with students proposing and rule that CS61C can hold 2 proposals, with the following class rankings:

CS61C	A > D > C > B
CS70	C > A > D > B
CS61A	D > A > B > C

Day	CS61C	CS70	CS61A
1	A, B, C	D	
2	A, C	D	B

Assignment: (CS61C, A and C), (CS70, D), (CS61A, B).

(d) Follow the stability proof in the lecture note to prove that the assignment is stable. Suppose some TA T in the assignment prefers some class C^* to their assigned class C. We will argue that C^* prefers their TA(s) to T, so there cannot be a rogue couple (a class and a TA prefer each other to their current assignments). Since C^* occurs before C in T's list, he must have proposed to it before he proposed to C. Therefore, according to the algorithm, C^* must have rejected him for somebody it prefers. If C^* is not CS61C, the Improvement Lemma shows C^* likes its final TA at least as much as T. If C^* is CS61C, then we must now prove an alternate Improvement Lemma to show this.

Prove: If T is rejected by CS61C on the k-th day, then every subsequent day $j \geq k$, CS61C has 2 TAs whom it likes at least as much as T.

- *Base case*: On day k, CS61C rejects T, so it must prefer the two TAs it holds.
- *Induction hypothesis*: Suppose claim is true for $j \geq k$
- *Induction step*: On day $j+1$, by induction hypothesis, CS61C has 2 TAs T' and T'' it prefers to T. Either nobody proposes to CS61C, or T''' proposes. If T''' is accepted, then it must be preferred over T' or T'', which are both at least as good as T, so T''' is preferred over T.

After proving the alternate Improvement Lemma, we can claim C^* likes its final TA at least as much as T. Therefore, no TA T can be involved in a rogue couple, and thus the assignment is stable.

6. (20 points:10/10) **(Better Off Alone)**

In the stable marriage problem, suppose that some men and women have standards and would not just settle for anyone. In other words, in addition to the preference orderings they have, they prefer being alone to being with some of the lower-ranked individuals (in their own preference list). A pairing could ultimately have to be partial, i.e., some individuals would remain single.

The notion of stability here should be adjusted a little bit. A pairing is stable if

- there is no paired individual who prefers being single over being with his/her current partner,
- there is no paired man and paired woman that would both prefer to be with each other over their current partners, and
- there is no single man and single woman that would both prefer to be with each other over being single.

(a) (10 points) Prove that a stable pairing still exists in the case where we allow single individuals. You can approach this by introducing imaginary mates that people "marry" if they are single. How should you adjust the preference lists of people, including those of the newly introduced imaginary ones for this to work?

Answer: Following the hint, we introduce an imaginary mate (let's call it a robot) for each person. Note that we introduce one robot for each individual person, i.e. there are as many robots as there are people. For simplicity let us say each robot is owned by the person we introduce it for.

Each robot is in love with its owner, i.e. it puts its owner at the top of its preference list. The rest of its preference list can be arbitrary. The owner of a robot puts it in his/her preference list exactly after the last person he/she is willing to marry. i.e. owners like their robots more than

people they are not willing to marry, but less than people they like to marry. The ordering of people who someone does not like to marry as well as robots he/she does not own is irrelevant as long as they all come after their robot.

To illustrate, consider this simple example: there are three men $1,2,3$ and three women A,B,C. The preference lists for men is given below:

Man	Preference List
1	$A > B$
2	$B > A > C$
3	C

and the following depicts the preference lists for women:

Woman	Preference List
A	1
B	$3 > 2 > 1$
C	$2 > 3 > 1$

In this example, 1 is willing to marry A and B and he likes A better than B, but he'd rather be single than to be with C. On the other side B has a low standard and does not like being single at all. She likes 3 first, then 2, then 1 and if there is no option left she is willing to be forced into singleness. On the other hand, A has pretty high standards. She either marries 1 or remains single.

According to our explanation we should introduce a robot for each person. Let's name the robot owned by person X as R_X. So we introduce male robots R_A, R_B, R_C and female robots R_1, R_2, R_3. Now we should modify the existing preference lists and also introduce the preference lists for robots.

According to our method, 1's preference list should begin with his original preference list, i.e. $A > B$. Then comes the robot owned by 1, i.e. R_1. The rest of the ordering, which should include C and R_2, R_3 does not matter, and can be arbitrary.

For B, the preference list should begin with $3 > 2 > 1$ and continue with R_B, but the ordering between the remaining robots (R_A and R_C) does not matter.

What about robots' preference lists? They should begin with their owners and the rest does not matter. So for example R_A's list should begin with A, but the rest of the humans/robots (B, C, R_1, R_2, and R_3) can come in any arbitrary order.

So the following is a list of preference lists that adhere to our method. There are arbitrary choices which are shown in bold (everything in bold can be reordered within the bold elements).

Man	Preference List
1	$A > B > R_1 > \mathbf{3} > \mathbf{R_3} > \mathbf{R_2}$
2	$B > A > C > R_2 > \mathbf{R_1} > \mathbf{R_3}$
3	$C > R_3 > \mathbf{R_1} > \mathbf{R_3} > \mathbf{A} > \mathbf{B}$
R_A	$A > \mathbf{B} > \mathbf{C} > \mathbf{R_1} > \mathbf{R_2} > \mathbf{R_3}$
R_B	$B > \mathbf{R_1} > \mathbf{R_2} > \mathbf{R_3} > \mathbf{A} > \mathbf{C}$
R_C	$C > \mathbf{A} > \mathbf{R_2} > \mathbf{B} > \mathbf{R_1} > \mathbf{R_3}$

and the following depicts the preference lists for women and female robots:

Woman	Preference List
A	$1 > R_A > \mathbf{3} > \mathbf{R_B} > \mathbf{2} > \mathbf{R_C}$
B	$3 > 2 > 1 > R_B > \mathbf{R_C} > \mathbf{R_A}$
C	$2 > 3 > 1 > R_C > \mathbf{R_A} > \mathbf{R_B}$
R_1	$1 > \mathbf{R_B} > \mathbf{2} > \mathbf{R_C} > \mathbf{3} > \mathbf{R_A}$
R_2	$2 > \mathbf{R_A} > \mathbf{R_C} > \mathbf{1} > \mathbf{3} > \mathbf{R_B}$
R_3	$3 > \mathbf{2} > \mathbf{1} > \mathbf{R_A} > \mathbf{R_C} > \mathbf{R_B}$

Now let us prove that a stable pairing between robots and owners actually corresponds to a stable pairing (with singleness as an option). This will finish the proof, since we know that in the robots and owners case, the propose and reject algorithm will give us a stable matching.

It is obvious that to extract a pairing without robots, we should simply remove all pairs in which there is at least one robot (two robots can marry each other, yes). Then each human who is not matched is declared to be single. It remains to check that this is a stable matching (in the new, modified sense). Before we do that, notice that a person will never be matched with another person's robot, because if that were so he/she and his/her robot would form a rogue couple (the robot's love is there, and the owner actually likes his/her robot more than other robots).

i. No one who is paired would rather break out of his/her pairing and be single. This is because if that were so, that person along with its robot would have formed a rogue couple in the original pairing. Remember, the robot loves its owner more than anything, so if the owner likes it more than his/her mate too, they would be a rogue couple.

ii. There is no rogue couple. If a rogue couple m and w existed, they would also be a rogue couple in the pairing which includes robots. If neither m nor w is single, this is fairly obvious. If one or both of them are single, they prefer the other person over being single, which in the robots scenario means they prefer being with each other over being with their robot(s) which is their actual match.

This shows that each stable pairing in the robots and humans setup gives us a stable pairing in the humans-only setup. It is noteworthy that the reverse direction also works. If there is a stable pairing in the humans-only setup, one can extend it to a pairing for robots and humans setup by first creating pairs of owners who are single and their robots, and then finding an arbitrary stable matching between the unmatched robots (i.e. we exclude everything other than the unmatched robots and find a stable pairing between them). To show why this works, we have to refute the possibility of a rogue pair. There are three cases:

i. A human-human rogue pair. This would also be rogue pair in the humans-only setup. The humans prefer each other over their current matches. If their matches are robots, that translates to them preferring each other over being single in the humans-only setup.

ii. A human-robot rogue pair. If the human is matched to his/her robot, our pair won't be a rogue pair since a human likes his/her robot more than any other robot. On the other hand if the human is matched to another human, he/she prefers being with that human over being single which places that human higher than any robot. Again this refutes the human-robot pair being rogue.

iii. A robot-robot rogue pair. If both robots are matched to other robots, then by our construction, this won't be a rogue couple (we explicitly selected a stable matching between left-alone robots). On the other hand, if either robot is matched to a human, that human is its owner, and obviously a robot loves its owner more than anything, including other robots. So again this cannot be a rogue pair.

This completes the proof.

(b) (10 points) As you saw in the lecture, we may have different stable pairings. But interestingly, if a person remains single in one stable pairing, s/he must remain single in any other stable pairing as well (there really is no hope for some people!). Prove this fact by contradiction.

Answer: We will perform proof by contradiction. Assume that there exists some man m_1 who is paired with a woman w_1 in stable pairing S and unpaired in stable pairing T. Since S is a stable pairing and m_1 is unpaired, w_1 must be paired in T with a man m_2 whom she prefers over m_1. (If w_1 were unpaired or paired with a man she does not prefer over m_1, then (m_1, w_1) would be a rouge couple, which is a contradiction.)

Since m_2 is paired with w_1 in T, he must be paired in S with some woman w_2 whom m_2 prefers over w_1. This process continues (w_2 must be paired with some m_3 in T, m_3 must be paired with some w_3 in S, etc.) until all persons are paired. Since this requires m_1 to be paired in T, where he is known to be unpaired, we have reached a contradiction. Therefore, our assumption must be false, and there cannot exist some man who is paired in a stable pairing S and unpaired in a stable pairing T. A similar argument can be used for women.

Since no man or woman can be paired in one stable pairing and unpaired in another, every man or woman must be either paired in all stable pairings or unpaired in all stable pairings.

Here is another possible proof:

We know that some male-optimal stable pairing exists. Call this pairing M. We first establish two lemmas.

Lemma 1. If a man is single in male-optimal pairing M, then he is single in all other stabling pairings.

Proof. Assume there exists a man that is single in M but not single in some other stable pairing M'. Then M would not be a male-optimal pairing, so this is a contradiction.

Lemma 2. If a woman is paired in male-optimal pairing M, she is paired in all other stable pairings.

Proof. Assume there exists a woman that is paired in M but single in some other stable pairing M'. Then M would not be female-pessimal, so this is a contradiction.

Let there be k single men in M. Let M' be some other stable pairing. Then by Lemma 1, we know single men in M' will be greater than or equal to k. We also know that there are $n-k$ paired men and women in M. Then by Lemma 2, we know that the number of paired women in M' will be greater than or equal to $n-k$.

Now, we want to prove that if a man is paired in M, then he is paired in every other stable pairing. We prove this by contradiction. Assume that there exists a man m that is paired in M but is single in some other stable pairing M'. Then there must be strictly greater than k single men in M', and thus strictly greater than k single women in M'. Since there are strictly greater than k single women in M', there must be strictly less than $n-k$ paired women in M'. But this contradicts that the number of paired women in M' will be greater than or equal to $n-k$.

We also have to prove that if a woman is single in M, then she must be single every other stable pairing. We again prove this by contradiction. Assume that there exists a woman w that is single in M and paired in some other stable pairing M'. Then there are strictly greater than $n-k$ paired women in M', which means there are strictly greater than $n-k$ paired men in M'. This means

there must be strictly less than k single men in M'. But this contradicts that the number of single men in M' will be greater than or equal to k.

Since we have proved both 1) If a man is single in M then he is single in every other stable pairing and 2) If a man is paired in M then he is paired in every other stable pairing (note that the contrapositive of this is if a man is single in any other stable pairing, then this man is single in M), we know that a man is single in M if and only if he is single in every other stable pairing. Similarly, since we have proved both 1) If a woman is single in M then she is single in every other stable pairing and 2) If a woman is paired in M then she is paired in every other stable pairing, we know that a woman is single in M if and only if she is single in every stable pairing. Thus we have proved that if a person is single in one stable pairing, s/he is single in every stable pairing.

CS 70 Discrete Mathematics and Probability Theory
Spring 2016 Rao and Walrand HW 3

Due Thursday February 11th at 10PM

1. **Homework process and study group** Who else did you work with on this homework? List names and student ID's. (In case of hw party, you can also just describe the group.) How did you work on this homework? Working in groups of 3-5 will earn credit for your "Sundry" grade.

2. **Stable Roommates Problem** (10 points)

   Suppose you are the head of the residential assistants in charge of assigning roommates. Currently you have four students who are going to move in: Kevin, Bryan, Jimmy and Michael. They have submitted their application with the following preferences:

Kevin	Bryan $>$ Michael $>$ Jimmy
Bryan	Jimmy $>$ Kevin $>$ Michael
Jimmy	Kevin $>$ Bryan $>$ Michael
Michael	Kevin $>$ Bryan $>$ Jimmy

   Consider an modified propose-and-reject algorithm which consists of each person x, one by one, proposing to the first person y on their list and executing as follows:

   - When y is proposed by x, y crosses off everyone below x on y's list.
   - If y holds 2 proposals, y rejects the person y prefers least (crosses off the person on y's list).
   - When x is rejected by y, x crosses off y on x's list and proposes to the next person immediately.

   This continues until everyone holds exactly one proposal. We start with the following proposals and produce the following table:

Kevin	$\rightarrow$	Bryan, Bryan crosses off Michael
Bryan	$\rightarrow$	Jimmy, Jimmy crosses off Michael
Jimmy	$\rightarrow$	Kevin
Michael	$\rightarrow$	Kevin, Kevin rejects/crosses off Jimmy, Jimmy crosses off Kevin
Jimmy	$\rightarrow$	Bryan, Bryan rejects/crosses off Kevin, Kevin crosses off Bryan
Kevin	$\rightarrow$	Michael, Michael crosses off Bryan and Jimmy

Kevin	~~Bryan~~ $>$ Michael $>$ ~~Jimmy~~
Bryan	Jimmy $>$ ~~Kevin~~ $>$ ~~Michael~~
Jimmy	~~Kevin~~ $>$ Bryan $>$ ~~Michael~~
Michael	Kevin $>$ ~~Bryan~~ $>$ ~~Jimmy~~

Since each person only has a list size of one, the algorithm terminates with the pairing: {(Kevin, Michael), (Bryan, Jimmy)}.

Now consider there are two more students who want to apply for housing: Joshua and Justin. Try the algorithm on the following table to find a pairing:

Kevin	Bryan	>	Joshua	>	Michael	>	Jimmy	>	Justin
Bryan	Joshua	>	Jimmy	>	Kevin	>	Justin	>	Michael
Jimmy	Kevin	>	Joshua	>	Bryan	>	Michael	>	Justin
Michael	Justin	>	Kevin	>	Bryan	>	Jimmy	>	Joshua
Joshua	Kevin	>	Justin	>	Bryan	>	Michael	>	Jimmy
Justin	Jimmy	>	Michael	>	Joshua	>	Kevin	>	Bryan

Note: The output of this example will be a stable pairing. However, for any instance, if it has a stable pairing, the algorithm cannot guarantee to find the stable pairing. In fact, the algorithm described above is only the Phase 1 of the Irving Algorithm. With the Phase 2, the Irving Algorithm can always find a stable pairing, if the given instance has one. For more information, please check https://en.wikipedia.org/wiki/Stable_roommates_problem

Answer:

Kevin	→	Bryan	Bryan crosses off Justin and Michael.
Bryan	→	Joshua	Joshua crosses off Michael and Jimmy.
Jimmy	→	Kevin	Kevin crosses off Justin.
Michael	→	Justin	Justin crosses off Joshua, Kevin, and Bryan.
Joshua	→	Kevin	Kevin crosses off Michael and Jimmy; Kevin rejects Jimmy.
			Jimmy crosses off Kevin.
Jimmy	→	Joshua	Joshua rejects Jimmy.
			Jimmy crosses off Joshua.
Jimmy	→	Bryan	Bryan crosses off Kevin; Bryan rejects Kevin.
			Kevin crosses off Bryan.
Kevin	→	Joshua	Joshua crosses off Justin and Bryan; Joshua rejects Bryan.
			Bryan crosses off Joshua.
Bryan	→	Jimmy	Jimmy crosses off Michael and Justin.
Justin	→	Jimmy	Jimmy rejects Justin.
			Justin crosses off Jimmy.
Justin	→	Michael	Michael crosses off Kevin, Bryan, Jimmy, and Joshua.

Kevin	~~Bryan~~	>	Joshua	>	~~Michael~~	>	~~Jimmy~~	>	~~Justin~~
Bryan	~~Joshua~~	>	Jimmy	>	~~Kevin~~	>	~~Justin~~	>	~~Michael~~
Jimmy	~~Kevin~~	>	~~Joshua~~	>	Bryan	>	~~Michael~~	>	~~Justin~~
Michael	Justin	>	~~Kevin~~	>	~~Bryan~~	>	~~Jimmy~~	>	~~Joshua~~
Joshua	Kevin	>	~~Justin~~	>	~~Bryan~~	>	~~Michael~~	>	~~Jimmy~~
Justin	~~Jimmy~~	>	Michael	>	~~Joshua~~	>	~~Kevin~~	>	~~Bryan~~

Pairing: (Kevin, Joshua), (Bryan, Jimmy), (Michael, Justin).

3. **Induction on Graphs** (5 points)

   What is wrong with the following "proof"?

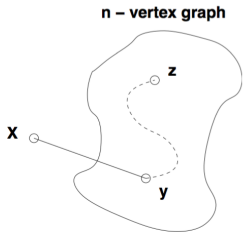
   False Claim: If every vertex in an undirected graph has degree at least 1, then the graph is connected.

Proof: We use induction on the number of vertices $n \geq 1$.

Base case: There is only one graph with a single vertex and it has degree 0. Therefore, the base case is vacuously true, since the if-part is false.

Inductive hypothesis: Assume the claim is true for some $n \geq 1$.

Inductive step: We prove the claim is also true for $n+1$. Consider an undirected graph on n vertices in which every vertex has degree at least 1. By the inductive hypothesis, this graph is connected. Now add one more vertex x to obtain a graph on $(n+1)$ vertices, as shown below.



All that remains is to check that there is a path from x to every other vertex z. Since x has degree at least 1, there is an edge from x to some other vertex; call it y. Thus, we can obtain a path from x to z by adjoining the edge $\{x,y\}$ to the path from y to z. This proves the claim for $n+1$. □

Answer: The mistake is in the argument that *"every $(n+1)$-vertex graph with minimum degree 1 can be obtained from an n-vertex graph with minimum degree 1 by adding 1 more vertex."* Instead of starting by considering an arbitrary $(n+1)$-vertex graph, this proof only considers an $(n+1)$-vertex graph that you can make by starting with an n-vertex graph with minimum degree 1. As a counterexample, consider a graph on four vertices $V = \{1,2,3,4\}$ with two edges $E = \{\{1,2\},\{3,4\}\}$. Every vertex in this graph has degree 1, but there is no way to build this 4-vertex graph from a 3-vertex graph with minimum degree 1.

More generally, this is an example of *build-up error* in proof by induction. Usually this arises from a faulty assumption that every size $n+1$ graph with some property can be "built up" from a size n graph with the same property. (This assumption is correct for some properties, but incorrect for others, such as the one in the argument above.)

One way to avoid an accidental build-up error is to use a *"shrink down, grow back"* process in the inductive step: start with a size $n+1$ graph, remove a vertex (or edge), apply the inductive hypothesis $P(n)$ to the smaller graph, and then add back the vertex (or edge) and argue that $P(n+1)$ holds.

Let's see what would have happened if we'd tried to prove the claim above by this method. In the inductive step, we must show that $P(n)$ implies $P(n+1)$ for all $n \geq 1$. Consider an $(n+1)$- vertex graph G in which every vertex has degree at least 1. Remove an arbitrary vertex v, leaving an n-vertex graph G' in which every vertex has degree... uh-oh! The reduced graph G' might contain a vertex of degree 0, making the inductive hypothesis $P(n)$ inapplicable! We are stuck —— and properly so, since the claim is false!

4. **Graphs**

   (a) (8 points) Suppose we have n websites such that for every pair of websites A and B, either A has a link to B or B has a link to A. Prove or disprove that there exists a website that is reachable from every other website by clicking at most 2 links. (*Hint: Induction*)

   Answer: We prove this by induction on the number of websites n.

   Base case For $n = 2$, there's always a link from one website to the other.

Induction Hypothesis When there are k websites, there exists a website w that is reachable from every other website by clicking at most 2 links.

Induction Step Let A be the set of websites with a link to w, and B be the set of websites two links away from w. The induction hypothesis states that the set of k websites $W = \{w\} \cup A \cup B$. Now suppose we add another website v. Between this website and every website in W, there must be a link from one to the other. If there is at least one link from v to $\{w\} \cup A$, w would still be reachable from v with at most 2 clicks. Otherwise, if all links from $\{w\} \cup A$ point to v, v will be reachable from every website in B with at most 2 clicks, because every website in B can click one link to go to a website in A, then click on one more link to go to v. In either case there exists a website in the new set of $k+1$ websites that is reachable from every other website by clicking at most 2 links.

(b) (8 points) We have shown in the lecture (or you have read Lecture Note 5) that a connected undirected graph has an Eulerian tour if and only if every vertex has even degree.

Prove or disprove that if a connected graph G on n vertices has exactly $2d$ vertices of odd degree, then there are d walks that *together* cover all the edges of G (i.e., each edge of G occurs in exactly one of the d walks; and each of the walks should not contain any particular edge more than once).

Answer: We split the $2d$ odd-degree vertices into d pairs, and join each pair with an edge, adding d more edges in total. Notice that now all vertices in this graph are of even degree. Now by Euler's theorem the resulting graph has an Eulerian tour. Removing the d added edges breaks the tour into d walks covering all the edges in the original graph, with each edge belonging to exactly one walk.

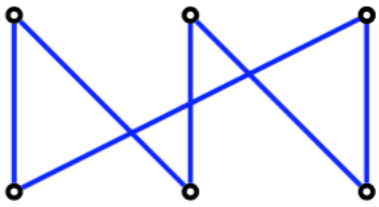
5. **Another Problem on Graphs**

In this problem, we are given a bipartite graph: $G = (L, R, E)$ where there are two sets of vertices, L and R, and $E \subseteq L \times R$, or each edge is incident to a vertex in L and a vertex in R. We also know that every vertex has degree *exactly* d.

We wish to partition the edges into d perfect matchings: a perfect matching is a set of edges where every vertex is incident to exactly one edge in the matching. Another view is that each vertex is matched to another vertex; similar to a pairing in stable marriage except that the pair must correspond to an edge in the graph. A matching is a set of edges where the number of edges incident to any vertex is at most 1 (as opposed to equal to 1 for a perfect matching.)

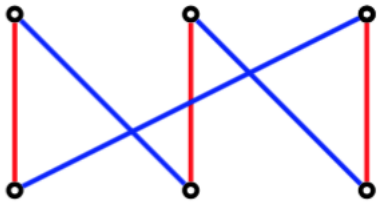
(a) (5 points) Draw a 6 vertex example graph that for $d = 2$ that meets the conditions above for an instance.

Answer:



(b) (5 points) Indicate two matchings in your graph that cover the edges.

Answer:



The red and blue edges each form a perfect matching.

(c) (6 points) Prove that for any instance of this problem that $|L| = |R|$. (Remember every vertex has degree d for any instance.)

Answer: Each edge is incident to one vertex in $|L|$ and in $|R|$, thus the total number of edge incidences with L and $|R|$ is $|E|$.

The edges incidences for L (R) is $d|L|$ ($d|R|$) by definition of degree. Thus,

$$d|L| = |E| = d|R| \implies |L| = |R|.$$

(d) (6 points) Prove that the length of any cycle in an instance of this problem is even.

Answer: Proof: Take a walk along a cycle, since each edge goes between V_0 and V_1, at each step, the set of the resulting vertex alternates in each step. Thus, to to return to the starting point as a cycle must, the alternation must occur an even number of times, and the cycle must have an even number of edges.

(e) (6 points) Prove that you can partition the edges in a simple cycle in this graph into exactly two perfect matchings with respect to the vertices in the cycle.

Answer: Proof: Take a walk along the cycle, and color each edge alternately 1 and 2. Each middle node is adjacent to exactly one edge colored 1 and one edge colored 2. The starting edge is colored 1 and the ending edge is colored two, thus the starting/ending vertex is also incident to only one edge of each color. The colors partition the edges into two sets, and each vertex has degree 1 in each set. Thus each set is a matching on the vertices in the cycle. □

(f) (6 points) Assume d is a power of 2; $d = 2^k$ for some natural number k. Give an efficient algorithm to compute a partition of the edges into perfect matchings. (Note that trying all possible partitions is not efficient. The algorithm should not take exponential time.)

Answer: Algorithm:: Find an eulerian tour in each connected component of the graph. Walk along the path coloring each edge with color 1 and color 2. Now, we recurse on the two degree $d/2$ graphs of color 1 and color 2 edges, and union the partitions of the edges in the two subgraphs.. If the graph has degree 1, we return all the games as the 1 week solution.

(g) (6 points) Prove your algorithm from the previous part is correct.

Answer: Proof: Each intermediate vertex is incident to $d/2$ edges of color 1, and $d/2$ edges of color 2. The starting vertex is also incident to $d/2$ of each color; when it is in the middle of the tour the incoming is one color, the outgoing is another, and the start/end edges are differently colored since any tour has even length.

We can inductively assume that the procedure produces a feasible partition into matchings on the two subgraphs as their degrees are a power of 2. Thus, we get a total of d perfect matchings.

The base case is degree 1, we return a single set of edges which clearly induces degree 1 on the vertices. □

6. **Trees** (10 points)

Show that the edges of a complete graph on n vertices for even n can be partitioned into $\frac{n}{2}$ edge disjoint spanning trees.

Recall that a complete graph is an undirected graph with an edge between every pair of vertices. The complete graph has $\frac{n(n-1)}{2}$ edges. A spanning tree is a tree on all n vertices — so it has $n-1$ edges. So the complete graph has enough edges (for even n) to create exactly $\frac{n}{2}$ edge disjoint spanning trees (i.e. each edge participates in exactly one spanning tree). You have to show that this is always possible.

Answer: We proceed by induction.

Base Case: Consider a complete graph on 2 vertices; this is simply •—•. This can clearly be partitioned into $\frac{2}{2} = 1$ edge disjoint spanning tree, because the graph is already a tree.

Inductive Hypothesis: Assume that the edges of a complete graph on k vertices (for k even) can be partitioned into $\frac{k}{2}$ edge disjoint spanning trees.

Inductive Step: We need to partition the edges of a complete graph G_{k+2} on $k+2$ vertices into $\frac{k}{2}+1$ edge disjoint spanning trees.

To do this, label the vertices of G_{k+2} as $v_1, v_2, \ldots, v_{k+2}$. Remove the vertices v_{k+1} and v_{k+2} (and associated edges) to form a complete graph G_k with k vertices $v_1, \ldots, v_k$. By the inductive hypothesis, G_k has $\frac{k}{2}$ edge disjoint spanning trees; call these trees $T_1, \ldots, T_{k/2}$.

Add the vertices v_{k+1} and v_{k+2} back into G_k to once again form the graph G_{k+2}. These vertices come with $2k+1$ extra edges, connecting (v_i, v_{k+1}) and (v_i, v_{k+2}) for each $i = 1, 2, \ldots, k$, and also (v_{k+1}, v_{k+2}). These edges must be included into spanning trees.

We wish to extend the trees $T_1, \ldots, T_{k/2}$ to include the new vertices v_{k+1} and v_{k+2}. To do this, for each tree T_i, attach two new edges (v_i, v_{k+1}) and $(v_{i+k/2}, v_{k+2})$. This extends each tree T_i to be a spanning tree.

The remaining edges form one additional spanning tree. These edges are $(v_{i+k/2}, v_{k+1})$ and (v_i, v_{k+2}) for $i = 1$ to $k/2$, along with the connecting edge (v_{k+1}, v_{k+2}). These edges connect each of the vertices v_{k+1} and v_{k+2} to half the remaining vertices, and together with the edge between v_{k+1} and v_{k+2} this gives the desired spanning tree.

Therefore, we have covered the graph in $\frac{k}{2}+1$ edge disjoint spanning trees. This completes the induction.

Remark: The key idea here is the following:

Take a graph with k vertices that is partitioned into $\frac{k}{2}$ spanning trees. In the inductive step, we want to add two vertices (with associated edges). To maintain a partitioning into spanning trees, we must expand the preexisting $\frac{k}{2}$ trees to the new vertices, but this is a bit subtle!

We need to add the two new vertices to each of the preexisting $\frac{k}{2}$ trees, which takes $2 \cdot \frac{k}{2} = k$ edges connecting the preexisting k vertices to the two new vertices. *It's really important that we use only one edge out of each of the original vertices!* This is because otherwise, we would use up both new edges out of one of the vertices v_j, but then our final new spanning tree wouldn't be able to reach v_j, so the remaining $k+1$ edges wouldn't be able to form a spanning tree!

So to do this, we need to split the original k vertices into two equal subsets of $\frac{k}{2}$ vertices each, and connect each half to one of the two new vertices. Once we do that, we can then justify forming a new spanning tree from the remaining edges, which allows us to complete the argument.

7. **Another Problem on Trees**

Recall that a **tree** is a connected acyclic graph (graph without cycles). In the note, we presented a few other definitions of a tree, and in this problem, we will prove two fundamental properties of a tree, and derive two definitions of a tree we learn from lecture note based on these properties. Let's start with the properties:

(a) (6 points) Prove that any pair of vertices in a tree are connected by exactly one (simple) path.

Answer: Pick any pair of vertices x, y. We know there is a path between them since the graph is connected. We will prove that this path is unique by contradiction:

Suppose there are two distinct paths from x to y. At some point (say at vertex a) the paths must diverge, and at some point (say at vertex b) they must reconnect. So by following the first path from a to b and the second path in reverse from b to a we get a cycle. This gives the necessary contradiction.

(b) (6 points) Prove that adding any edge to a tree creates a simple cycle.

Answer: Pick any pair of vertices x, y not connected by an edge. We prove that adding the edge $\{x,y\}$ will create a simple cycle. From part (a), we know that there is a unique path between x and y. Therefore, adding the edge $\{x,y\}$ creates a simple cycle obtained by following the path from x to y, then following the edge $\{x,y\}$ from y back to x.

Now you will show that if a graph satisfies either of these two properties then it must be a tree:

(c) (6 points) Prove that if every pair of vertices in a graph are connected by exactly one simple path, then the graph must be a tree.

Answer: Assume we have a graph with the property that there is a unique simple path between every pair of vertices. We will show that the graph is a tree, namely, it is connected and acyclic. First, the graph is connected because every pair of vertices is connected by a path. Moreover, the graph is acyclic because there is a unique path between every pair of vertices. More explicitly, if the graph has a cycle, then for any two vertices x, y in the cycle there are at least two simple paths between them (obtained by going from x to y through the right or left half of the cycle), contradicting the uniqueness of the path. Therefore, we conclude the graph is a tree.

(d) (6 points) Prove that if the graph has no simple cycles and has the property that the addition of any single edge (not already in the graph) will create a simple cycle, then the graph is a tree.

Answer: Assume we have a graph with no simple cycles, but adding any edge will create a simple cycle. We will show that the graph is a tree. We know the graph is acyclic because it has no simple cycles. To show the graph is connected, we prove that any pair of vertices x, y are connected by a path. We consider two cases: If $\{x,y\}$ is an edge, then clearly there is a path from x to y. Otherwise, if $\{x,y\}$ is not an edge, then by assumption, adding the edge $\{x,y\}$ will create a simple cycle. This means there is a simple path from x to y obtained by removing the edge $\{x,y\}$ from this cycle. Therefore, we conclude the graph is a tree.

8. **Hypercubes**

(a) (10 points) Prove that any cycle in an n-dimensional hypercube must have even length.

Recall that a cycle is a closed (simple) path and its length is the number of vertices (edges) in it. The n dimensional hypercube is a graph whose vertex set is the set of n-bit strings, with an edge between vertices u, v iff they differ in exactly one bit (Hamming distance $= 1$).

Answer: Here are three ways to solve this problem: argue via bit flips, parity of Hamming distance, or induction on n. In each case we try to give credit to solutions according to how clearly they expressed the main idea. However, induction on n is more difficult and prone to build-up error.

Answer 1: Bit flips

Main idea: moving through an edge in a hypercube flips exactly one bit, and moreover each bit must be flipped an even number of times to end up at the starting vertex of the cycle.

Here are a sequence of four proofs roughly based on this idea, starting with the most convincing and ending with the least convincing. Also included is a critique saying what is missing in the later proofs.

Proof 1: Each edge of the hypercube flips exactly one bit position. Let E_i be the set of edges in the cycle that flip bit i. Then $|E_i|$ must be even. This is because bit i must be restored to its original value as we traverse the cycle, which means that bit i must be flipped an even number of times. Since each edge of the cycle must be in exactly one set E_j, the total number of edges in the cycle = $\sum_j |E_j|$ is a sum of even numbers and therefore even.

Proof 2: Let C be a cycle in an n-dimensional hypercube. As we go along the edges of C we must end up where we started. Because traversing an edge in a hypercube flips exactly one bit, this means every flipped bit must eventually be flipped back. This means that the number of edges in C must be even. □

Proof 3: Each edge of the cycle flips one bit. Let the starting point be x, and let the farthest the cycle goes from x be y which is at a Hamming distance of k. Then the cycle must flip all those k bits back to return to x. Therefore the total number of edges in the cycle is $k+k = 2k$, an even number.

Proof 4: By induction on n, the dimension of the hypercube. For the induction step, we know that the $(n+1)$-dimensional hypercube is made up of two n-dimensional hypercubes, where every vertex in one n-dimensional hypercube has an edge connected to their 'twin' vertex in the other n-dimensional hypercube. Any cycle in the $(n+1)$-dimensional hypercube has to go back and forth from one n-dimensional hypercube to the other an even number of times, since otherwise it will start in one n-dimensional hypercube, and end in the other, and cannot be a cycle. So each edge in the cycle is either in n-dimensional hypercube or the other or goes between the two n-dimensional hypercubes. The number of edges of each of the first two types is even by the induction hypothesis and the last number is even as shown earlier. Therefore the total number of edges in the cycle is even. □

Critique of Proofs 2–4:

i. Proof 2 is almost correct, and got full credit. However, if one were to be picky one might say that the proof does not make it clear that each bit can be flipped back and forth, and must indeed flip an even number of times (which might be greater than 2) to return to its original value. The proof is also not very explicit about stating that the total number of edges in the cycle is even because the number of edges in the cycle corresponding to each bit position is even.

ii. Proof 3 claims the total length of the cycle is twice the Hamming distance from the starting point x to the farthest point on the cycle. But this claim is false. Starting from x the cycle can move farther from and closer to x many times, and moreover if y is the farthest point in Hamming distance from x, then the number of edges from x to y in the cycle does not need to be equal to the number of edges from y back to x.

iii. Proof 4 is even farther from being a proof. It correctly points out that the number of times the cycle crosses back and forth between the two n-dimensional cubes must be even (this is just saying that the number of times bit 1 is flipped is even). But then it appeals to the inductive hypothesis, and this is quite meaningless, since the intersection of the cycle with the n-dimensional hypercube will not in general look anything like a cycle.

Answer 2: Parity

Main idea: Parity of Hamming distance is equal to parity of number of edges traversed, so to get a cycle we need an even number of edges.

Sample proof: Let C be a cycle in an n-dimensional hypercube, and let x be any vertex in C. Argue (using induction on k or other techniques) that if we start from x and walk for k edges to another vertex y, then the Hamming distance $H(x,y)$ is even if and only if k is even. Each edge traversal brings about a change in vertex binary representation at one bit. Therefore, the Hamming distance changes by 1 and its parity flips when we traverse an edge.
But as we go along the edges of C we must eventually get back to x. At this point the Hamming distance is $H(x,x) = 0$, which is even. This shows that the number of edges in the cycle must be even as well. □

Answer 3: Induction on n

The most common answer made the mistake of a pretty serious build-up error. Here is an example of such a proof:

Proof by induction on n, the dimension of the hypercube.

Base case: For $n = 2$, there is only one cycle and it has length 4, which is even.
Induction Hypothesis: Any cycle in a n-dimensional hypercube has even length.
Induction Step: Let C be a cycle in the $(n+1)$-dimensional hypercube. The $(n+1)$-dimensional hypercube is made up of two n-dimension hypercubes. There are two cases:

i. C lies in one of the two n-dimensional hypercubes. In this case we are done by the induction hypothesis.
ii. C crosses between the two n-dimensional hypercubes. In this case there is an even cycle in each of the two n-dimensional hypercubes. To connect them, we must remove an edge from each of the two cycles in the n-dimensional hypercubes and connect each of the endpoints to their twin vertex in the other n-dimensional hypercube. Now the number of edges in the cycle is odd + odd + 2 = even, where odd = even cycle − one edge.

□

There are many problems with this proof. First, it completely ignores the fact that the cycle can go back and forth between the two n-dimensional hypercubes a number of times. But even if we were to focus on the special case where it goes back and forth just once, the proof is still seriously wrong. This is because even in this case, the part of the cycle in each n-dimensional hypercube is just a path between two possibly distant vertices, i.e., it need not look anything like a cycle with one edge deleted. In particular this path could be of even or odd length.

There is nonetheless a way of writing down a correct proof by induction. This involves a couple of ideas, including strengthening the induction hypothesis to any tour in the n-dimensional hypercube:

Main idea: A tour in an n-dimensional hypercube can be decomposed into some components in the two $(n-1)$-dimensional subcubes plus an even number of crossing edges. The components in both subcubes can be superimposed to form a tour in the $(n-1)$-dimensional hypercube, allowing us to apply the inductive hypothesis.

Sample proof: We use induction on n.
Base case: For $n = 2$, it is easy to show that every tour has even length.
Inductive hypothesis: Any tour in the $(n-1)$-dimensional hypercube has even length.
Inductive step: Let C be a tour in the n-dimensional hypercube. Consider the decomposition of the n-dimensional hypercube into two $(n-1)$-dimensional subcubes. We decompose C into three parts: the edges that lie in the first subcube, the edges in the second subcube, and the edges crossing the subcubes. Argue that because C is a tour, the number of crossing edges must be even (but not necessarily 2). The edges of C in each subcube do not have to be a tour; in fact they can be collections of disjoint paths. The components in one subcube also don't have to be equal or symmetrical to the components in the other subcube. But argue that when you superimpose them (superimpose the vertices of one subcube with the corresponding vertices in the other subcube), you get a tour in the $(n-1)$-dimensional hypercube. Now apply the inductive hypothesis to conclude that the total number of edges of C in both subcubes must be even. To get the total number of edges in C we need to add the number of crossing edges, which is also even. This completes the inductive step. □

(b) (10 points) A *Hamiltonian path* in an undirected graph $G = (V, E)$ is a path that goes through every vertex *exactly once*. A *Hamiltonian cycle* (or *Hamiltonian tour*) is a cycle that goes through every vertex exactly once. Note that, in a graph with n vertices, a Hamiltonian path consists of $n-1$ edges, and a Hamiltonian cycle consists of n edges.

Prove that for every $n \geq 2$, the n-dimensional hypercube has a Hamiltonian cycle.

Answer:

We proceed by induction on n. In the base case $n = 2$, we have the 2-dimensional hypercube, which is a square graph on $V = \{00, 01, 10, 11\}$. Here we have a Hamiltonian cycle $00 \to 01 \to 11 \to 10 \to 00$.

Suppose now that the $(n-1)$-dimensional hypercube has a Hamiltonian cycle. Let $v \in \{0,1\}^{n-1}$ be a vertex adjacent to 0^{n-1} (the notation 0^{n-1} means a sequence of $n-1$ zeroes) in the Hamiltonian cycle. By removing the edge $\{0^{n-1}, v\}$ from the cycle, we obtain a Hamiltonian path in the $(n-1)$-dimensional hypercube that starts at 0^{n-1} and ends at v.

We now want to construct a Hamiltonian cycle in the n-dimensional hypercube. Recall the decomposition of the n-dimensional hypercubes into 0-subcube and 1-subcube, where the 0-subcube (respectively, the 1-subcube) is the $(n-1)$-dimensional hypercube with vertices labeled by $0x$ (respectively, $1x$) for $x \in \{0,1\}^{n-1}$, and every vertex $0x$ in the 0-subcube is connected to the corresponding vertex $1x$ in the 1-subcube.

Then the following is a Hamiltonian cycle in an n-dimensional hypercube: have a path that goes from $0^n \in \{0,1\}^n$ to $0v$ by passing through all vertices in the 0-subcube (this is simply a copy of the Hamiltonian path in dimension $(n-1)$ from 0^{n-1} to v), then an edge from $0v$ to $1v$, then a path from $1v$ to 10^{n-1} that passes through all vertices in the 1-subcube (this is another copy of the Hamiltonian path in dimension $(n-1)$ traveled in reverse), and finally an edge from 10^{n-1} to 0^n. This completes the proof.

9. **Four Colorable?** (10 points)

In the lecture, we have shown that every planar graph can be colored with five colors. We have also shown that it can also be colored with only four colors. The coloring example of U.S. map is shown below. In this question, prove the following: any planar graph of maximum degree 4 has a four coloring.

Answer: We proceed with a proof by induction on the number of vertices n. $P(n)$ denotes that any planar graph of maximum degree 4 has a four coloring.

Base case: $P(n)$ is trivially true for $n <= 4$.

Inductive Step: Assume $P(n1)$ is true: planar graph with $n1$ vertices of maximum degree 4 has a four coloring. Now consider a planar graph with n vertices of maximum degree 4.

Remove an arbitrary vertex v and its incident edges from the graph. The graph with $n1$ vertices still has maximum degree 4 so it's four colorable by hypothesis. Now we try to add vertex v back to the colored graph with $n-1$ vertices. If there is an unused color from all of the neighbors of v, then we can assign an unused color to v and the graph is still four colorable.

Now let's consider the case where v has degree four and every other color is used for its neighbors. For each neighbor of v, we label them clockwise in order v_1,v_2,v_3,v_4, containing corresponding colors 1, 2, 3, 4. Let's consider v_1 and v_3. Say we attempt to change the color of v_1 from 1 to 3. If these two vertices aren't connected, then we can recolor v_1 to color 3 and assign v to color 1. We are done. In the case where v_1 and v_3 are in fact connected, let's consider vertices v_2 and v_4. These two vertices cannot be connected because if a path between those two vertices would have to cut through v_1 and v_3. It can't happen due to the planar graph. We can then assign the color 2 or 4 to vertex v and make v_2 and v_4 have the same color. This makes $P(n)$ true, completing the inductive step.

CS 70 Discrete Mathematics and Probability Theory
Spring 2016 Rao and Walrand HW 4

Due Thursday 18th at 10PM

1. **Amaze your friends!**

   (a) You want to trick your friends into thinking you can perform mental arithmetic with very large numbers What are the last digits of the following numbers?

   i. 11^{2014}
   ii. 9^{10001}
   iii. $3^{987654321}$

   (b) You know that you can quickly tell a number n is divisible by 9 if and only if the sum of the digits of n is divisible by 9. Prove that you can use this trick to quickly calculate if a number is divisible by 9.

   Answer:

   Motivation for Problem: This problem causes students to start recognizing tricks regarding modular arithmetic. This lays the ground later for proving properties of modular arithmetic.

   Solutions:

   (a) i. 11 is always 1 mod 10 therefore the answer to (a) is 1.
   ii. 9 is its own inverse mod 10, therefore, if 9 is raised to an odd power, the number will be 9 mod 10. So the answer is 9
   iii. $3^4 = 9^2 = 1 \bmod 10$. We see that the exponent $987654321 = 1 \bmod 4$ so the answer is 3.

   (b) Let n be written as $a_k a_{k-1} \cdots a_1 a_0$ where the a_i are digits, base-10. We can write
   $n = 10^k a_k + 10^{k-1} a_{k-1} + \cdots + 10a_1 + a_0 = (10^k - 1)a_k + (10^{k-1} - 1)a_{k-1} + \cdots + (10-1)a_1 + \sum_{i=0}^{k} a_i$
   The first few terms are all divisible 9; they're all of the form $99\cdots99 \cdot a_i$. So if the sum at the end is divisible by 9, then n is too and vice versa.

2. **Short Answer: Modular Arithmetic**

   (a) What is the multiplicative inverse of $3 \pmod 7$?

   Answer: $5 \pmod 7$.
   $(3)(5) = 15 = 1 \pmod 7$

   (b) What is the multiplicative inverse of $n-1$ modulo n? (An expression that may involve n. Simplicity matters.)

   Answer: $n-1 \pmod n$.
   Its $-1 \pmod n$! Or $(n-1)(n-1) = n^2 - 2n + 1 = 1 \pmod n$.

(c) What is the solution to the equation $3x = 6 \pmod{17}$? (A number in $\{0,\ldots,16\}$ or "No solution".)

Answer: 2.
Muliply both sides by 6 the multiplicative inverse of 3 and reduce.

(d) Let $R_0 = 0; R_1 = 2; R_n = 4R_{n-1} - 3R_{n-2}$ for $n \geq 2$. Is $R_n = 2 \pmod{3}$ for $n \geq 1$? (True or False)

Answer: True.
Take the recursive formula modulo 3. This is a warmup question for the next problem.

(e) Given that $extended - gcd(53, m) = (1, 7, -1)$, that is $(7)(53) + (-1)m = 1$, what is the solution to $53x + 3 = 10 \pmod{m}$? (Answer should be an expression that is interpreted $\pmod{m}$, and shouldn't consists of fractions.)

Answer: $x = 49 \pmod{m}$
Follows from 7 being multiplicative inverse of $53 \pmod{m}$.

3. **(Combining moduli)** Suppose we wish to work modulo $n = 40$. Note that $40 = 5 \times 8$, with $\gcd(5, 8) = 1$. We will show that in many ways working modulo 40 is the same as working modulo 5 and modulo 8, in the sense that instead of writing down $c \pmod{40}$, we can just write down $c \pmod{5}$ and $c \pmod{8}$.

(a) What is $8 \pmod{5}$ and $8 \pmod{8}$? Find a number $a \pmod{40}$ such that $a \equiv 1 \pmod{5}$ and $a \equiv 0 \pmod{8}$.

Answer: $8 \equiv 3 (\text{mod } 5)$ and $8 \equiv 0 (\text{mod } 8)$. We can find such a number by considering multiples of 8, i.e. 0, 8, 16, 24, 32, and find that if $a = 16$, $16 \equiv 1 (\text{mod } 5)$. Therefore 16 satisfies both conditions.

(b) Now find a number $b \pmod{40}$ such that $b \equiv 0 \pmod{5}$ and $b \equiv 1 \pmod{8}$.

Answer: We can find such a number by considering multiples of 5,i.e.0,5,10,15,20,25,30,35, and find that if $b = 25$, $25 \equiv 1 (\text{mod } 8)$, so it satisfies both conditions.

(c) Now suppose you wish to find a number $c \pmod{40}$ such that $c \equiv 2 \pmod{5}$ and $c \equiv 5 \pmod{8}$. Find c by expressing it in terms of a and b.

Answer: We claim $c \equiv 2a + 5b \equiv 37 (\text{mod } 40)$. To see that $c \equiv 2 (\text{mod } 5)$, we note that $b \equiv 0 (\text{mod } 5)$ and $a \equiv 1 (\text{mod } 5)$. So $c \equiv 2a \equiv 2 (\text{mod } 5)$. Similarly $c \equiv 5b \equiv 5 (\text{mod } 8)$.

(d) Repeat to find a number $d \pmod{40}$ such that $d \equiv 3 \pmod{5}$ and $d \equiv 4 \pmod{8}$.

Answer: We can repeat the same procedure as above, and find that $d = 3a + 4b \equiv 28 (\text{mod } 40)$.

(e) Compute $c \times d \pmod{40}$. Is it true that $c \times d \equiv 2 \times 3 \pmod{5}$, and $c \times d \equiv 5 \times 4 \pmod{8}$?

Answer: $c \times d = 37 \times 28 \equiv 36 \pmod{40}$. Note that if $a \equiv b \pmod{n}$ and $c \equiv d \pmod{n}$ then $a \times c \equiv b \times d \pmod{n}$.Therefore we can multiply $c \equiv 2 \pmod{5}$ and $d \equiv 3 \pmod{5}$ to get

$c \times d \equiv 2 \times 3 \pmod 5$. Similarly we can multiply these equations (mod 8) and get $c \times d = 5 \times 4 \pmod 8$.

4. **(The last digit)**
Let a be a positive integer. Consider the following sequence of numbers x defined by:

$$\begin{aligned} x_0 &= a \\ x_n &= x_{n-1}^2 + x_{n-1} + 1 \text{ if } n > 0 \end{aligned}$$

(a) Show that if the last digit of a is 3 or 7, then for every n, the last digit of x_n is respectively 3 or 7.

Answer: To answer this question, we can study how the last digit of x_n changes from n to $n+1$. We have the following table:

$x_n \bmod 10$	$x_{n+1} \bmod 10$
0	1
1	3
2	7
3	3
4	1
5	1
6	3
7	7
8	3
9	1

(b) Show that there exist $k > 0$ such that the last digit of x_n for $n \geq k$ is constant. Give the smallest possible k,
no matter what a is. **Answer:** 3 and 7 appear as our fixed points. Once we reach one of these, we stay there for all the following iterations by the previous question. But it is not immediate that we always reach one of the fixed points, and this is what we need to prove. Let's unroll each of the 10 cases for a for a few iterations and verify that we always reach 3 or 7.

$a \bmod 10$	$x_1, x_2, \ldots \bmod 10$
0	1, 3, 3, 3, ...
1	3, 3, 3, ...
2	7, 7, 7, ...
3	3, 3, 3, ...
4	1, 3, 3, 3, ...
5	1, 3, 3, 3, ...
6	3, 3, 3, 3, ...
7	7, 7, 7, 7, ...
8	3, 3, 3, 3, ...
9	1, 3, 3, 3, ...

This case-splitting proves the claim. We can see from the table that $k = 2$ is the smallest constant such that the last digit of x_n is constant for $n \geq k$.

5. (a) Compute the inverse of 37 modulo 64 using Euclid's extended GCD algorithm.
Answer:

We can use the following form to find the inverse using Euclid's extended GCD algorithm, and the x,y for this case would be 64 and 37 since we need to have $x \geq y \geq 0$;

x, y	d	a, b
64, 37	1	11, -19
37, 27	1	-8, 11
27, 10	1	3, -8
10, 7	1	-2, 3
7, 3	1	1, -2
3, 1	1	0, 1
1, 0	1	1, 0

Here's how to read the chart:
The LHS top down is just the standard GCD algorithm, the last row indicates where we find the GCD for 64 and 37, which is 1. Then the RHS (including the middle column) bottom up is the recursive return value for extended GCD algorithm. Finally, the a,b value (11,-19) in the top row will be the return value for extended-gcd(64,37). We can check that this pair is indeed the value we are looking for by calculating $11 * 64 - 19 * 37 = 1$ i.e. $a * x + b * y = 1$
Therefore, the inverse of 37 modulo 64 is -19.

(b) Prove that $gcd(F_n, F_{n-1}) = 1$, where $F_0 = 0$ and $F_1 = 1$ and $F_n = F_{n-1} + F_{n-2}$.
Answer: We prove this by induction.

In the base case, we have $gcd(F_1, F_0) = gcd(1, 0) = 1$, which is trivially true.
Inductive hypothesis: Assume we have $gcd(F_k, F_{k-1}) = 1$ for some $k \geq 1$
Inductive steps: Now we need to show that $gcd(F_{k+1}, F_k) = 1$ as well.
We can show that:
$gcd(F_{k+1}, F_k) = gcd(F_k + F_{k-1}, F_k) = gcd(F_k, (F_k + F_{k-1}) - F_k) = gcd(F_k, F_{k-1}) = 1$
Therefore the statement is also true for $n = k + 1$.
By the rule of induction, we can conclude that $gcd(F_n, F_{n-1}) = 1$ for all $n \geq 1$ where $F_0 = 0$ and $F_1 = 1$ and $F_n = F_{n-1} + F_{n-2}$.

6. **Bijections**

Let n be an odd number. Let $f(x)$ be a function from $\{0, 1, \ldots, n-1\}$ to $\{0, 1, \ldots, n-1\}$. In each of these cases say whether or not $f(x)$ is a bijection. Justify your answer (either prove $f(x)$ is a bijection or give a counterexample).

(a) $f(x) = 2x \pmod{n}$.

Answer: Bijection, because there exists the inverse function $g(y) = 2^{-1}y \pmod{n}$
(See Lemma 7.1 from Lecture note 7). Since n is odd, $gcd(2, n) = 1$, so the multiplicative inverse of 2 exists (See Theorem 6.2 from Lecture note 6).

(b) $f(x) = 5x \pmod{n}$.

Answer: Not a bijection. For example, $n = 5, f(0) = f(1) = 0$.

(c) n is prime and

$$f(x) = \begin{cases} 0 & \text{if } x = 0 \\ x^{-1} \pmod{n} & \text{if } x \neq 0 \end{cases}$$

Answer: Bijection, because the multiplicative inverse is unique (Theorem 6.2).

(d) n is prime and $f(x) = x^2 \pmod{n}$.

Answer: Not a bijection. For example, if n = 3, f (1) = f (2) = 1.

7. Using RSA (8 points, 5/3)

Kevin and Bob decide to apply the RSA cryptography so that Kevin can send a secret message to Bob.

(a) Assuming $p = 3$, $q = 11$, and $e = 7$, what is d? Calculate the exact value.

Answer: $(3-1)(11-1) = 20$, so d is the multiplicative inverse of 7 mod 20. Run egcd$(20,7)$ and get $1 = (-1) \times 20 + (3) \times 7$, so $d = 3$.
Note: You can also try $d = 1, 2, 3, \ldots$ and get $d = 3$.

(b) Following Part (a), what is the original message if Bob receives 4? Calculate the exact value.

Answer: $N = 3 \times 11 = 33$. $4^d = 4^3 = 64 \equiv 31 \pmod{33}$.

8. **Tweaking RSA**

(This problem will not be graded, the solution will be posted on the problem thread on piazza.)

(a) You are trying to send a message to your friend, and as usual, Eve is trying to decipher what the message is. However, you get lazy, so you use $N = p$, and p is prime. Similar to the original method, for any message $x \in \{0, 1, \ldots, N-1\}$, $E(x) \equiv x^e \pmod{N}$, and $D(y) \equiv y^d \pmod{N}$. Show how you choose e and d in the encryption and decryption function, respectively. Prove that the message x is recovered after it goes through your new encryption and decryption functions, $E(x)$ and $D(y)$.

Answer: Choose e such that it is coprime with $p-1$, and choose $d \equiv e^{-1} \pmod{p-1}$.
We want to show x is recovered by $E(x)$ and $D(y)$, such that $D(E(x)) = x$.
In other words, $x^{ed} \equiv x \pmod{p}$ $\forall x \in \{0, 1, \ldots, N-1\}$.
Proof: By construction of d, we know that $ed \equiv 1 \pmod{p-1}$. This means we can write $ed = k(p-1)+1$, for some integer k, and $x^{ed} = x^{k(p-1)+1}$.

- x is a multiple of p: Then this means $x = 0$, and indeed, $x^{ed} \equiv 0 \pmod{p}$.
- x is not a multiple of p: Then $x^{ed} \equiv x^{k(p-1)+1} \equiv x^{k(p-1)}x \equiv 1^k x \equiv x \pmod{p}$, by using FLT.

And for both cases, we have shown that x is recovered by $E(D(y))$.

(b) Can Eve now compute d in the decryption function? If so, by what algorithm?

Answer: Since Eve knows $N = p$, and $d \equiv e^{-1} \pmod{p-1}$, now she can compute d using EGCD.

(c) Now you wonder if you can modify the RSA encryption method to work with three primes ($N = pqr$ where p, q, r are all prime). Explain how you can do so.

Answer: Let e be co-prime with $(p-1)(q-1)(r-1)$. Give the public key: (N, e) and calculate $d = e^{-1} \bmod (p-1)(q-1)(r-1)$. People who wish to send me a secret, x, send $y = x^e \bmod N$. We decrypt an incoming message, y, by calculating $y^d \bmod N$.

Does this work? We prove that $x^{ed} - x \equiv 0 \bmod N$, and thus $x^{ed} = x \bmod N$.
To prove that $x^{ed} - x \equiv 0 \bmod N$, we factor out the x to get
$x \cdot (x^{ed-1} - 1) = x \cdot (x^{k(p-1)(q-1)(r-1)+1-1} - 1)$ because $ed \equiv 1 \bmod (p-1)(q-1)(r-1)$.
We now show that $x \cdot (x^{k(p-1)(q-1)(r-1)} - 1)$ is divisible by p, q, and r. Thus, it is divisible by N, and $x^{ed} - x \equiv 0 \bmod N$.
To prove that it is divisible by p:

- if x is divisible by p, then the entire thing is divisible by p.
- if x is not divisible by p, then that means we can use FLT on the inside to show that $(x^{p-1})^{k(q-1)(r-1)} - 1 \equiv 1 - 1 \equiv 0 \bmod p$. Thus it is divisible by p.

To prove that it is divisible by q:

- if x is divisible by q, then the entire thing is divisible by q.
- if x is not divisible by q, then that means we can use FLT on the inside to show that $(x^{q-1})^{k(p-1)(r-1)} - 1 \equiv 1 - 1 \equiv 0 \bmod q$. Thus it is divisible by q.

To prove that it is divisible by r:

- if x is divisible by r, then the entire thing is divisible by r.
- if x is not divisible by r, then that means we can use FLT on the inside to show that $(x^{r-1})^{k(p-1)(q-1)} - 1 \equiv 1 - 1 \equiv 0 \bmod r$. Thus it is divisible by r.

CS 70 Discrete Mathematics and Probability Theory
Spring 2016 Rao and Walrand HW 5

Due Thursday 18th at 10PM

1. **Proof practice**

   The purpose of this problem is to practice formally proving a statement, when you intuitivelly "know" why it's true.

   Suppose that there are n chickens in a farmyard. Chickens are rather aggressive birds that tend to establish dominance in relationships by pecking. (Hence the term "pecking order".) In particular, for each pair of distinct chickens, either the first pecks the second or the second pecks the first, but not both. We say that chicken u virtually pecks chicken v if either:

   - Chicken u directly pecks chicken v, or
   - Chicken u pecks some other chicken w who in turn pecks chicken v.

   A chicken that virtually pecks every other chicken is called a *king chicken*.

   We can model this situation with a tournament digraph. The vertices are chickens, and an edge $u \to v$ indicates that chicken u pecks chicken v. Notice that there could be multiple kings.

   Theorem 1. *The chicken with the largest outdegree in an n-chicken tournament is a king.*

   Intuitively the theorem is true because if the chicken with the largest outdegree was not a king then there would be a chicken that pecks everyone that the fake king pecks, as well as the fake king, i.e. have an even larger outdegree.

   Turn this intuition into a formal proof.

   Answer:

   By contradiction, let u^* be the chicken with the largest outdegree, and assume it is not a king. Let $X = \{v|(u^*,v) \in E\}$ be the set of chicken that are pecked by u^*, and $Y = \{v|(x,v) \in E, x \in X\}$ the set of chicken that are pecked by chickens in X. The outdegree of u^* is equal to $|X|$.

   Let z be a chicken such that $z \notin X \cup Y$. This implies that $(v,z) \notin E$, for all $v \in X \cup \{u^*\}$. But, by the tournament property, this implies that $(z,v) \in E$, for all $v \in X \cup \{u^*\}$. Therefore the degree of z is equal to $|X|+1 > |X|$, a contradiction, since u^* has the highest outdegree.

2. **What's in a googolplex?**

   A "googolplex", the namesake of Google's "Googleplex" headquarters, is the number written as 1 followed by 10^{100} zeroes. That is, it's $10^{10^{100}}$. For a positive integer n, we define "n factorial" (written $n!$) as the product of all positive integers from 1 to n, i.e. $n \cdot (n-1) \cdot \ldots \cdot 2 \cdot 1$. As you might imagine, $10^{10^{100}}!$, googleplex factorial, is a Very Large Number. Let:

   $$\begin{aligned} m \;=\; & 5185925986735423542444467237838883862213453463463453456 2 \\ & 4343453463442098164524359134591810007513111459435723459 1 \\ & 1023509823745752342345759111744904220352511745677798903 1. \end{aligned}$$

Note that that's a single 168-digit number (it just doesn't fit onto one line). Calculate what $10^{10^{100}}!$ is congruent to, modulo m. That is, find the value of:

$$10^{10^{100}}! \mod m$$

Show all your work. Hint: if your answer takes more than 5-10 lines of text, you are probably doing it wrong.

Answer: The key observation is that, since m has 168 digits, $m < 10^{169} < 10^{10^{100}}$. Thus m must be one of the factors comprising googleplex factorial by definition, which means that googleplex factorial must be divisible by m and hence congruent to 0 modulo m. If you want to be exceedingly formal about it:

$$\begin{aligned}
10^{10^{100}}! \mod m &= \left(\prod_{i=1}^{10^{10^{100}}} i\right) \mod m \\
&= \left(\prod_{i=1}^{10^{10^{100}}} (i \mod m)\right) \mod m \\
&= \left(\prod_{i=1}^{m-1} (i \mod m)\right) \cdot (m \mod m) \cdot \left(\prod_{i=m+1}^{10^{10^{100}}} (i \mod m)\right) \mod m \\
&= \left(\prod_{i=1}^{m-1} (i \mod m)\right) \cdot 0 \cdot \left(\prod_{i=m+1}^{10^{10^{100}}} (i \mod m)\right) \mod m \\
&= 0
\end{aligned}$$

3. **(Polynomial Interpolations)**

   (a) Consider the set of four points $\{(0,1),(1,-2),(3,4),(4,0)\}$, construct the unique degree-3 polynomial (over the reals) that passes through these four points by writing down and solving a system of linear equations.

   Answer: Suppose the unique degree 3 polynomial passing through the four given points is

   $$p(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3$$

   The coefficients of $p(x)$ satisfy the following linear equations:

   $$p(0) = 1 \Rightarrow a_0 = 1 \tag{1}$$
   $$p(1) = -2 \Rightarrow a_0 + a_1 + a_2 + a_3 = -2 \tag{2}$$
   $$p(3) = 4 \Rightarrow a_0 + 3a_1 + 9a_2 + 27a_3 = 4 \tag{3}$$
   $$p(4) = 0 \Rightarrow a_0 + 4a_1 + 16a_2 + 64a_3 = 0 \tag{4}$$

   Substituting $a_0 = 1$ and subtracting (i) 3 times equation (2) from equation (3) and (ii) 4 times equation (2) from equation (4), we obtain the following simultaneous equations:

   $$6a_2 + 24a_3 = 12$$
   $$12a_2 + 60a_3 = 11$$

Solving for a_2 and a_3, we obtain $a_2 = 76/12$ and $a_3 = -13/12$. Substituting in equation (2) we obtain $a_1 = -99/12$. Hence

$$p(x) = (12 - 99x + 76x^2 - 13x^3)/12$$

is the unique degree 3 polynomial passing through the given points.

(b) Use Lagrange interpolation to find a polynomial $p(x)$ of degree at most 2 that passes through the points $(1,2)$, $(2,3)$, and $(3,5)$, working in $GF(7)$. In other words, we want $p(x)$ to satisfy $p(1) \equiv 2 \pmod 7$, $p(2) \equiv 3 \pmod 7$, and $p(3) \equiv 5 \pmod 7$. Show your work clearly and use the same notations as in Lecture Note 8.

Answer: First we would find the three $\Delta_i(x)$ polynomials.

$$\Delta_1(x) \equiv \frac{(x-2)(x-3)}{(1-2)(1-3)} \equiv \frac{(x-2)(x-3)}{2} \equiv 4(x-2)(x-3) \equiv 4x^2 + x + 3 \pmod 7$$
$$\Delta_2(x) \equiv \frac{(x-1)(x-3)}{(2-1)(2-3)} \equiv \frac{(x-1)(x-3)}{-1} \equiv -(x-1)(x-3) \equiv 6x^2 + 4x + 4 \pmod 7$$
$$\Delta_3(x) \equiv \frac{(x-1)(x-2)}{(3-1)(3-2)} \equiv \frac{(x-1)(x-2)}{2} \equiv 4(x-1)(x-2) \equiv 4x^2 + 2x + 1 \pmod 7$$

Next we compute $p(x)$:

$$\begin{aligned} p(x) &\equiv 2\Delta_1(x) + 3\Delta_2(x) + 5\Delta_3(x) \pmod 7 \\ &\equiv 2(4x^2 + x + 3) + 3(6x^2 + 4x + 4) + 5(4x^2 + 2x + 1) \pmod 7 \\ &\equiv x^2 + 2x + 6 + 4x^2 + 5x + 5 + 6x^2 + 3x + 5 \pmod 7 \\ &\equiv 4x^2 + 3x + 2 \pmod 7 \end{aligned}$$

4. **Secret Sharing**

Suppose we wish to share a secret among five people, and we decide to work modulo 7. We construct a degree-two polynomial $q(x) = ax^2 + bx + s$ by picking the coefficients a and b at random (mod 7); the constant term is the secret s (also a number mod 7). We give shares $q(1), \ldots, q(5)$ to each of the five people (all operations being done mod 7). Now suppose that three of the people get together and share the information that $q(1) = 5$, $q(2) = 2$, and $q(4) = 2$. Use Lagrange interpolation to find the polynomial q and the secret s. Show all your work.

Answer: For convenience, we will first list the inverse pairs modulo 7: $(1,1), (2,4), (3,5), (6,6)$. Now, to find a polynomial q such that $q(1) = 5$, $q(2) = 2$, and $q(4) = 2$, we must compute

$$q(x) = 5\Delta_1(x) + 2\Delta_2(x) + 2\Delta_4(x),$$

where each Δ_i is computed as follows:

$$\begin{aligned} \Delta_1 &= \frac{(x-2)(x-4)}{(1-2)(1-4)} = \frac{x^2 - 6x + 8}{(-1)(-3)} = 5(x^2 + x + 1) = 5x^2 + 5x + 5 \\ \Delta_2 &= \frac{(x-1)(x-4)}{(2-1)(2-4)} = \frac{x^2 - 5x + 4}{(1)(-2)} = 3(x^2 + 2x + 4) = 3x^2 + 6x + 5 \\ \Delta_4 &= \frac{(x-1)(x-2)}{(4-1)(4-2)} = \frac{x^2 - 3x + 2}{(3)(2)} = 6(x^2 + 4x + 2) = 6x^2 + 3x + 5 \end{aligned}$$

Substituting, we now have

$$\begin{aligned} q(x) &= 5(5x^2+5x+5)+2(3x^2+6x+5)+2(6x^2+3x+5) \\ &= (4x^2+4x+4)+(6x^2+5x+3)+(5x^2+6x+3) \\ &= x^2+x+3 \end{aligned}$$

5. **Properties of $GF(p)$**

   (a) Show that, if $p(x)$ and $q(x)$ are polynomials over the reals (or complex, or rationals) and $p(x) \cdot q(x) = 0$ for all x, then either $p(x) = 0$ for all x or $q(x) = 0$ for all x or both.

   Answer: We will show the contrapositive. Suppose that $p(x)$ and $q(x)$ are both non-zero polynomials of degree d_p and d_q respectively. Then $p(x) = 0$ for at most d_p values of x and $q(x) = 0$ for at most d_q values of x. Since there are an infinite number of values for x (because we are using complex, real, or rational numbers) we can always find an x, call it $x_{notzero!}$, for which $p(x_{notzero!}) \neq 0$ and $q(x_{notzero!}) \neq 0$. This gives us $p(x_{notzero!}) \cdot q(x_{notzero}) \neq 0$, so pq is non-zero.

   (b) Show that the claim in part (a) is false for finite fields $GF(p)$.

   Answer: In $GF(p)$, $x^{p-1}-1$ and x are both non zero polynomials, but when p is prime, their product $(x^p - x)$ is zero for all x by Fermat's little Theorem.
   Examples for a specific p are also acceptable. For example for $GF(2)$, $p(x) = x$ and $q(x) = x+1$ work.

CS 70 Discrete Mathematics and Probability Theory
Spring 2016 Rao and Walrand HW 6

Due Thursday March 3rd at 10PM

Warmup

1. **(2/2/2/2/2) Reviewing Lagrange**

   (a) Prove the following: If p is a prime and $y_1, \ldots, y_n \in \mathbb{N}$ are all different from 0 modulo p, then $y_1 \times \cdots \times y_n$ is also different from 0 modulo p.

   Answer: Since $y_1, \ldots, y_n \in \mathbb{N}$ are all different from 0 modulo p, p does not factor any of them. Thus the prime factorization of $y_1 \times \cdots \times y_n$ does not include p, so p does not divide $y_1 \times \cdots \times y_n$. So

   $$y_1 \times \cdots \times y_n \not\equiv 0 \pmod{p}.$$

   (b) Prove the following: Given a prime p and two integers a, b, it is always possible to find a polynomial $f(x)$ of degree at most 1 such that $f(0) \equiv a \pmod{p}$ and $f(1) \equiv b \pmod{p}$.

   Answer: Let $f(x) = a + (b-a)x$. Then $f(0) = a \equiv a \pmod{p}$ and $f(1) = a + (b-a) = b \equiv b \pmod{p}$.

   (c) You are given a prime p and a positive number $n < p$. Show how to find a polynomial $f(x)$ of degree at most n satisfying $f(0) \equiv f(1) \equiv \cdots \equiv f(n-1) \equiv 0 \pmod{p}$ and $f(n) \equiv 1 \pmod{p}$. In other words, the polynomial f should be congruent to zero at the points $x = 0, \ldots, n-1$; at $x = n$ the polynomial should be 1 mod p.

   Hint: Consider $F(x) = (x-0)(x-1)(x-2)\cdots(x-(n-1))$; what can you say about it?

   Answer: Let $F(x) = (x-0)(x-1)(x-2)\cdots(x-(n-1))$, and define $a = F(n) \bmod p$. Note that $a \equiv n! \pmod{p}$ is invertible modulo p, since $n < p$ (we have $a^{-1} \equiv n^{-1} \times (n-1)^{-1} \times \cdots \times 1^{-1} \pmod{p}$, and each of $1, \ldots, n$ are invertible modulo p since they are less than p and thus relatively prime to p). Let $b = a^{-1} \bmod p$. Therefore, we may take

   $$f(x) = bF(x).$$

   We will have $f(0) \equiv f(1) \equiv \cdots \equiv f(n-1) \equiv 0 \pmod{p}$, since $F(0) = \cdots = F(n-1) = 0$. Also, we will have $f(n) \equiv F(n)^{-1}F(n) \equiv 1 \pmod{p}$. Finally, F has degree n since it has n terms in its definition, and so this choice of f has degree at most n. Consequently, this choice of f satisfies all the requirements.

   (d) You are given p and n as before, but now you are also given an index j with $0 \le j \le n$. Show how to find a polynomial $g_j(x)$ of degree at most n satisfying

   $$g_j(i) \equiv \begin{cases} 0 & \text{if } i \neq j \\ 1 & \text{if } i = j \end{cases} \pmod{p} \qquad \text{for each } i = 0, 1, \ldots, n.$$

In other words, the polynomial g_j should be congruent to zero at the points $x = 0, \ldots, n$, except that at $x = j$ it should be congruent to 1 mod p.

Answer: Use the same idea as in part (c). Define $G_j(x) = (x-0)\cdots(x-(j-1))(x-(j+1))\cdots(x-n)$ and $g_j(x) = (G_j(j)^{-1} \bmod p)G_j(x)$. As before, $g_j(i) = 0$ if $i \neq 0$, $g_j(j) \equiv 1 \pmod p$, and g_j has degree at most n, so this satisfies all the requirements.

(e) You are given a prime p, a number n with $0 < n < p$, and a sequence of values $a_0, a_1, \ldots, a_n \pmod p$. Describe an efficient algorithm to find a polynomial $h(x)$ of degree at most n satisfying $h(0) \equiv a_0 \pmod p$, $h(1) \equiv a_1 \pmod p$, $\ldots$, $h(n) \equiv a_n \pmod p$.

Hint: What can you say about the polynomial $3g_0(x) + 7g_1(x)$, where $g_0(x), g_1(x)$ are as defined in part (d)? Does this give you any ideas?

Answer: Let

$$h(x) = a_0 g_0(x) + a_1 g_1(x) + \cdots + a_n g_n(x),$$

with the g_j's defined as above. Then h has degree at most n, since the g_j's do, and moreover $h(i) \equiv a_0 g_0(i) + \cdots + a_n g_n(i) \equiv 0 + \cdots + 0 + a_i \cdot 1 + 0 + \cdots + 0 \equiv a_i \pmod p$, as desired.

Note that this h may be computed efficiently. Multiplying a polynomial of degree d by $(x-i)$ modulo p requires d multiplications and d additions modulo p, so we can compute each $G_j(x)$ in $O(n^2(\lg p)^2)$ time. Inverting $G_j(j)$ modulo p can be done in $O((\lg p)^3)$ time, as we saw in class. Thus we can compute all the g_j's in $O(n^3(\lg p)^2 + n(\lg p)^3)$ time, and then multiplying by the a_i's and adding gives us a total runtime of $O(n^3(\lg p)^2 + n^2(\lg p)^2 + n(\lg p)^3)$ to compute h modulo p. Since the input is $n \lg p$ bits long, this shows that the running time of the algorithm is polynomial in the input size (in fact, at worst cubic) and thus can be considered efficient.

Polynomials

2. **(2/2/2/2/2) Representing Polynomials**

Let f be a polynomial of degree at most d. The *coefficient representation* of f is the sequence $(a_0, a_1, \ldots, a_d)$ of coefficients of f. A *point-value representation* of f is a collection $\{(x_1, f(x_1)), (x_2, f(x_2)), \ldots, (x_t, f(x_t))\}$ of the values of f at any t points $x_1, x_2, \ldots, x_t$, where $t \geq d+1$. (Recall from Lecture Note 7 that a polynomial of degree d is completely determined by its values at any $d+1$ points. Note that t may be greater than $d+1$, so more points than necessary may be given.)

In the following questions, let f and g by any two real polynomials of degree at most d.

(a) What is the maximum degree of the product polynomial fg?

(b) Given coefficient representations of f and g, explain how to compute the coefficient representation of fg using $O(d^2)$ arithmetic operations (additions/subtractions/multiplications/divisions) over real numbers.

(c) Now suppose that f and g are specified by point-value representations at t points for some $t \geq d+1$, i.e., f is specified as $(x_1, f(x_1)), (x_2, f(x_2)), \ldots, (x_t, f(x_t))$, and g as $(x_1, g(x_1)), (x_2, g(x_2)), \ldots, (x_t, g(x_t))$. With a suitable value of t (which you should specify), show how to compute a point-value representation of fg using only $O(d)$ arithmetic operations.

(d) Suppose that polynomial g divides polynomial f, and that f, g are given in point-value representation as in part (c) with $t = d+1$. Show how to compute a point-value representation for the quotient f/g using $O(d)$ arithmetic operations, and justify your algorithm carefully.

(e) Suppose you are given f in coefficient representation, and you want to compute a point-value representation for f at $t = d+1$ points. Show how to do this using $O(d^2)$ arithmetic operations. [HINT: Show how to evaluate f at one point using $O(d)$ operations; to do this, consider writing f in the form $f(x) = a_0 + xh(x)$, where h is a polynomial of degree at most $d-1$, and iterating.]

Answer:

(a) Suppose the polynomial $f(x) = a_0 + a_1x + \cdots a_kx^k$ and $g(x) = b_0 + b_1x + \cdots b_lx^l$ where k and l are at most d. The coefficient c_s of x^s in the product polynomial fg is given by:

$$c_s = a_0b_s + a_1b_{s-1} + \cdots + a_sb_0 = \sum_{i+j=s} a_ib_j \tag{1}$$

The expression shows that the coefficient c_s equals 0 if $s > k+l$. The maximum possible degree of fg is therefore $k+l \leq 2d$.

(b) The expression (3) for the coefficients of fg shows that c_s can be computed with at most $s+1$ multiplications and s additions. The number of arithmetic operations to compute one coefficient of fg is at most $2(s+1) = O(d)$ as $s \leq 2d$ by part (a). There are at most $2d+1$ non-zero coefficients c_s, so all the coefficients of fg can be computed with $O(d^2)$ arithmetic operations.

(c) Given the tuples $(x_i, f(x_i))$ and $(x_i, g(x_i))$, using one multiplication we can compute $fg(x_i) = f(x_i) \cdot g(x_i)$. Using part (a), it is sufficient to have $2d+1$ tuples in a point-value representation for fg. Hence, provided that $t \geq 2d+1$, a point-value representation of fg can be computed with $2d+1 = O(d)$ multiplications.

(d) Given tuples $(x_i, f(x_i))$ and $(x_i, g(x_i))$, using one division we can compute $[f/g](x_i) = f(x_i)/g(x_i)$ if $g(x_i) \neq 0$. The computation is invalid if and only if the point x_i is a root of g. The number of roots of g is at most $deg(g)$, so $f(x_i)/g(x_i)$ can be evaluated using division on at least $d+1-deg(g)$ points. As the polynomial f is divisible by g we have $d+1-deg(g) \geq deg(f)+1-deg(g) = 1+deg(f/g)$.

Thus, a point-value representation for f/g can be found using $deg(f/g)+1 = O(d)$ divisions by computing $[f/g](x_i) = f(x_i)/g(x_i)$ for all x_i such that $g(x_i) \neq 0$.

(e) We note that $f(x) = a_0 + a_1x + \cdots + a_kx^k = a_0 + x.(a_1 + \cdots a_kx^{k-1})$, showing that a degree-$k$ polynomial can be written as $f(x) = a_0 + x.h(x)$ where the degree of $h(x)$ is $k-1$. It follows that a degree-k polynomial can be evaluated at point x by evaluating a polynomial of degree $k-1$ at x along with two additional arithmetic operations.

The number of arithmetic operations $A(k)$ required to evaluate a degree k polynomial therefore satisfies the recurrence $A(k) = 2 + A(k-1)$. Since a degree-0 polynomial a_0 can be evaluated with zero operations we have $A(0) = 0$, and by induction on k it follows that $A_k = 2k$.

Finally, computing a point-value representation for a polynomial of degree d requires evaluating the polynomial at $d+1$ points. The number of arithmetic operations required is therefore $2d(d+1) = O(d^2)$.

Error Correcting Codes

3. **(5/5) Error-correcting codes: an optimization**

In class, we saw an error-correcting code where the n message packets $m_1, \ldots, m_n$ are encoded to the $n+k$ encoded packets $c_1, \ldots, c_{n+k}$ by setting $P(x) = m_nx^{n-1} + \cdots + m_2x + m_1$, then defining $c_i = P(i)$

(all this is in $GF(q)$, where q is prime and larger than $n+k$, so each packet is a number in the range $0 \ldots q-1$). However, one possible criticism of this error-correcting code is that decoding always requires a Lagrange interpolation step, even if no packets are lost.

(a) In this part, you will develop a scheme that addresses this criticism. Let's preserve the basic approach where $c_i = Q(i)$ (for $i = 1, 2, \ldots, n+k$), for some appropriately chosen polynomial $Q(x)$ which encodes the entire message, and which has degree at most $n-1$. (As before, we'll work in $GF(q)$, where $q > n+k$ and q is prime.) At the same time, let's ensure $c_1 = m_1$, $c_2 = m_2$, ..., $c_n = m_n$, so that if no packets are lost, we can just use the first n encoded packets to immediately read off the message. Describe how to choose $Q(x)$ with this desired property, given $m_1, \ldots, m_n$. In other words, describe an efficient algorithm we can use for encoding.

Answer: Since $c_1 = m_1 = Q(1), \ldots, c_n = m_n = Q(n)$, we can efficiently construct $Q(x)$ using Lagrange interpolation. Specifically,

$$Q(x) = m_1\Delta_1(x) + m_2\Delta_2(x) + \cdots + m_n\Delta_n(x).$$

(b) For your scheme from part 1, if some packets are lost, the recipient can use Lagrange interpolation to recover $Q(x)$. Describe how the recipient could recover $m_1, \ldots, m_n$ from $Q(x)$.

Answer: Since $Q(1) = c_1 = m_1, \ldots, Q(n) = c_n = m_n$, we can recover the message $m_1, \ldots, m_n$ from $Q(x)$ simply by evaluating it at $1, 2, \ldots, n$ respectively.

4. **(5/5) List decoding**

(a) Consider a n character message encoded into m characters over the field $GF(p)$ using polynomials. Consider that one recieves $n-1$ of the m packets. Give a method to find a list of size at most p of possible messages. Your running time must be polynomial in terms of p, m, and n.

(b) Consider a n character message encoded into $m = n+2k$ characters over the field $GF(p)$ using polynomials. Consider that $k+1$ of the m received packets are corrupted. Give a method to find a list of possible messages which contains the original message. What is the size of the list for your scheme. It should be a small polynomial in p. Your running time must be polynomial in terms of p, m, and n.

Answer:

(a) Since we are trying to encode an n character message using polynomials, we are going to fit our message into a degree $n-1$ polynomial and then encode our message into the length m message $[P(0)P(1)\ldots P(m-1)]$. Now, we receive $n-1$ of these characters; suppose without loss of generality that the character at position k, $P(k)$ was not received. Now, we know $n-1$ points of the polynomial $P(x)$ - but knowing these $n-1$ points gives us no information about $P(k)$ since it is possible for us to construct a degree n polynomial that goes through the $n-1$ known points no matter what the value of $P(k)$ is. However, suppose we fix the value of $P(k)$ - then it turns out that there is exactly one polynomial that goes through the $n-1$ known points and $P(k)$, since a degree $n-1$ polynomial is uniquely determined by n of its points. We use this to deduce that there are at most p different polynomials $P(x)$ that could possibly be our encoding polynomial, one for each possible value of $P(k)$ - and thus there are at most p different possible messages that were originally sent. We would generate the p possible messages in the same way:

```
foreach value of l in the range 0 to p-1
    P(x) = interpolate(n-1 known points, P(k) = l)
    generate the possible message [P(0) P(1) ... P(m-1)]
end
```

(b) We can use a similar approach to above; we know that we have $k+1$; if we knew in advance that we were going to have $k+1$ errors we would have sent $n+2k+2$ packets in order to make sure that we could perform error correction using the Berlekamp-Welsh method to decode to the correct message. However, we ended up only sending $n+2k$ packets. If we knew the (correct) values of two more packets, then we could use Berlekamp-Welsh to decode the message. Since we do not know the values of two more packets, we can do what we did in part (a) and just guess what they are to generate possible messages. Since for the value of each packets there are p possible values, at most we will generate p^2 possible messages, and the real message will be included.

```
foreach value of a in the range 0 to p-1
     foreach value of b in the range 0 to p-1
         Use Berlekamp-Welsh with the known values and R(n + 2k + 1) = a
         and R(n + 2k + 2) = b
     end
end
```

5. **(3/3/4) Reed-Solomon and Reliable Computation**

In this question, we will see how error correction can help with faulty computations. Let us first establish two useful facts.

a) For a communication system that uses Reed-Solomon codes, what is the minimum number of additional packets to transmit for an intended message of n packets, knowing that the communication channel will corrupt at most a fraction $0 \leq f < \frac{1}{2}$ of the packets? (e.g. If $f = \frac{1}{4}$, that means that 3 out of 4 transmitted packets will be received flawlessly, but one quarter of them might have their contents corrupted in an arbitrary manner.)

Answer: A Reed-Solomon code which accounts for k general errors with intended message length n is $n+2k$ packets long. By definition, if the channel corrupts at most a fraction f of the packets, then there are at most $(n+2k)f$ corrupted packets. In order to account for these corrupted packets, we must have $k \geq (n+2k)f$, which we can rewrite as:

$$k \geq \frac{nf}{1-2f}$$

And hence, we need at least $2 \times \lfloor \frac{nf}{1-2f} \rfloor$ additional packets along with the intended n packets. Notice that if the channel corrupts more than half of the transmitted packets, no Reed-Solomon code can correct this.

b) Suppose we are using a Reed-Solomon code over $GF(p)$ guarding for k transmission errors. Let $a = (a_1, \ldots, a_n)$ and $b = (b_1, \ldots, b_n)$ be two n-packet messages. Show that the Reed-Solomon codeword for message $a+b = (a_1+b_1, \ldots, a_n+b_n)$ is the same as the sum of the Reed-Solomon codewords of a and b. In other words, the RS codeword of the element-wise sum is the element-wise sum of the RS codewords.

Answer: Let A, B and C be the message polynomials associated with messages a, b and $a+b$ respectively. By definition, A, B and C are of degree at most $n-1$ and are such that $A(i) = a_i$,

$B(i) = b_i$ and $C(i) = a_i + b_i$ for i in range $1, \ldots, n$. We have to show that $C(i) = A(i) + B(i)$ for all i in range $1, \ldots, n+2k$.

First, we have $C(i) = a_i + b_i = A(i) + B(i)$ for $1 \leq i \leq n$. What about the remaining $2k$ points? $A + B - C$ is a polynomial of degree at most $n-1$ with n distinct roots $1, \ldots, n$. Hence $A + B - C$ must be the zero polynomial ($A + B - C = 0$) which we can rewrite as $C = A + B$, thus finishing the proof.

Suppose you have invented a machine for doing additions extremely fast. Your invention takes a list of pairs of numbers as input and returns the list of the pairs sums. Although the machine is blazing fast, it is at the same time prone to mistakes. Luckily, you can bound the number of mistakes: over all of the n outputs returned, you know that at most $\max(1, \lfloor n/4 \rfloor)$ outputs have an error. For example, if we feed the machine $((2,3),(4,3),(0,7),(4,2))$ we might get back output $(5,7,4,6)$, where $4 \neq 0+7$ is a mistake.

You want to sell your invention, but none of your potential clients is interested in an error-prone device like this. They feel the speed benefit does not compensate for the unreliability of the results.

c) Show that you can augment your machine with a Reed-Solomon encoding and decoding scheme such that no wrong outputs are ever returned. Your clients can use the machine the exact same way as before, but they no longer experience erroneous results. More specifically, you need to define functions $E : GF(p)^n \to GF(p)^m$ and $D : GF(p)^m \to GF(p)^n$ such that $D(M(E(a_1, ..., a_n), E(b_1, ..., b_n)) = (a_1 + b_1, ..., a_n + b_n)$, even if M makes errors on up to $1/4$ of its additions.

Answer: Let n the number of input pairs $(a_1, b_1), \ldots, (a_n, b_n)$. The idea is to first encode $a = (a_1, \ldots, a_n)$ into a' and $b = (b_1, \ldots, b_n)$ into b' using an $(n, 2k)$-Reed-Solomon code, feed the original machine with the encoded pairs $(a'_1, b'_1), \ldots, (a'_{n+2k}, b'_{n+2k})$ to get the faulty summations, and then decode the obtained message and return the error-free solution. We are indeed guaranteed to get back the actual summations $(a_1 + b_1, \ldots, a_n + b_n)$ by question b) which shows that the sum of the Reed-Solomon codes corresponds to the Reed-Solomon code of the sums. Now, the only question is to find the number of errors k we need to account for in the Reed-Solomon scheme.

We have to be careful about the max condition in the error bound. There is at least 1 error independently of the input size n, which means we need $k \geq 1$. Furthermore, when $n \geq 4$ at most a fraction $\frac{1}{4}$ of the packets can get corrupted. Hence, by question a), we need:

$$k \geq \frac{n\frac{1}{4}}{1 - 2\frac{1}{4}} = \frac{n}{2}$$

when $n \geq 4$. Combining these two constraints and ignoring the $n \geq 4$ condition (which does not impact the correctness), we get the following bound:

$$k \geq \max(1, \frac{n}{2})$$

so we can use $2\max(1, \lfloor \frac{n}{2} \rfloor)$ additional inputs to make the scheme work.

Secret Sharing

6. **(3/3/4) Secret Sharing Pirate**

After a long and illustrious career as a buccaneer, Captain Flint passed away in the year 1754. He had split the gold accumulated over years of terrorizing ships into two batches, and just before his death he told his five faithful pirates the locations of the the two batches using a secret sharing scheme.

The captain chose polynomials $P(x)$ and $Q(x)$ over $GF(7)$, of degrees 1 and 2 respectively, with the secrets being the values $P(0)$ and $Q(0)$. For $1 \leq i \leq 5$, Pirate i received the two numbers $P(i)$ and $Q(i)$, but was not told which was which. The pirates cursed the Captain as they could not figure out how to recover the secrets, and the treasure lay undiscovered for many years.

On the eve of the 10th anniversary of the Captain's demise, the pirates captured a small vessel and encountered Monsieur Lagrange (who, having forsaken the ennui of land-life in favor of the vicissitudes of the life on the high seas, was himself an aspiring pirate). Lagrange offered his mathematical prowess in solving the pirates' problems, in exchange for a share of the treasure.

The secret shares received by the five pirates were $\{0,5\},\{1,4\},\{3,4\},\{0,4\}$ and $\{0,3\}$ respectively. (So, for example, Pirate 2's share was $\{1,4\}$, meaning that either $P(2) = 1$ and $Q(2) = 4$, or $P(2) = 4$ and $Q(4) = 1$.) Trace the following steps to see how M. Lagrange helped the pirates to solve the mystery.

(a) Find $P(0) + Q(0)$.

(b) Find $P(0)Q(0)$.

(c) Using parts (a) and (b), find the two secrets $P(0)$ and $Q(0)$ that were hidden by Captain Flint, assuming that $P(0) < Q(0)$.

Answer:

Summary: The sum $P(0) + Q(0)$ and the product $P(0) \cdot Q(0)$ of the secrets are recovered using Lagrange interpolation to find the polynomials $P(x) + Q(x)$ and $P(x) \cdot Q(x)$ from the given data. The secrets are obtained from the sum and product by solving a quadratic equation.

(a) $P(x) + Q(x)$ is a polynomial of degree 2 and passes through the points $(i, P(i) + Q(i))$. *(Note that we know all of these points exactly from the given data, because to compute $P(i) + Q(i)$ we don't need to know which value is $P(i)$ and which is $Q(i)$.)* We use the three points $(1,5),(2,5),(3,0)$ and write the Lagrange interpolation formula:

$$P(x) + Q(x) = 5 \cdot \Delta_1(x) + 5 \cdot \Delta_2(x) + 0 \cdot \Delta_3(x) \tag{2}$$

The polynomials Δ_i are given by:

$$\begin{aligned} \Delta_1(x) &= \frac{(x-2)(x-3)}{(1-2)(1-3)} = 4(x^2 - 5x + 6) \\ \Delta_2(x) &= \frac{(x-1)(x-3)}{(2-1)(2-3)} = 6(x^2 - 4x + 3) \end{aligned}$$

We compute $P(x) + Q(x)$ by substituting the expressions for Δ_i in the Lagrange interpolation formula (2):

$$\begin{aligned} P(x) + Q(x) &= 5 \cdot \Delta_1(x) + 5 \cdot \Delta_2(x) \\ &= -1(x^2 - 5x + 6) + 2(x^2 - 4x + 3) = x^2 - 3x \end{aligned}$$

The sum of the secrets $P(0) + Q(0)$ is therefore equal to 0.

(b) $P(x) \cdot Q(x)$ is a polynomial of degree 3 and passes through the points $(i, P(i) \cdot Q(i))$. We use the four points $(1,0),(2,4),(4,0),(5,0)$ and write the Lagrange interpolation formula:

$$P(x) \cdot Q(x) = 0 \cdot \Delta_1(x) + 4 \cdot \Delta_2(x) + 0 \cdot \Delta_4(x) + 0 \cdot \Delta_5(x) \tag{3}$$

Since three of these coefficients are zero, it is sufficient to compute the polynomial Δ_2:

$$\Delta_2(x) = \frac{(x-1)(x-4)(x-5)}{(2-1)(2-4)(2-5)} = 6(x^3 - 3x^2 + x + 1)$$

We compute $P(x) \cdot Q(x)$ by substituting the expression for Δ_2 in the Lagrange interpolation formula (3):

$$\begin{aligned} P(x) \cdot Q(x) &= 4 \cdot \Delta_2(x) \\ &= 3x^3 + 5x^2 + 3x + 3 \end{aligned}$$

The product of the secrets $P(0) \cdot Q(0)$ is therefore equal to 3.

(c) The square of the difference of the secrets is given by $(Q(0) - P(0))^2 = (P(0) + Q(0))^2 - 4 \cdot P(0) \cdot Q(0) = 2$. We observe that the two solutions to $x^2 = 2 \mod 7$ are ± 3 hence $Q(0) - P(0) = \pm 3 \mod 7$. Given that $Q(0) > P(0)$ we find that the secrets are $P(0) = 2$ and $Q(0) = 5$.

Counting

7. **(3/3/4) Counting Subsets**

Consider the set S of all (possibly infinite) subsets of $\mathbb{N}$.

(a) Show that there is a bijection between S and $T = \{f : \mathbb{N} \to \{0,1\}\}$ (the set of all functions that map each natural number to 0 or 1).

(b) Prove or disprove: S is countable.

Answer: Uncountable. Note that such f can be viewed as a binary encoding of a real number between 0 and 1, which exhibits a surjection from V to $[0,1]$.

Let T be a subset of $\mathbb{N}$. Define $f(T) = |T| + \sum_{t \in T} t$. Note that $f(T) \leq (|T|+1)\bar{t}$, where $\bar{t}$ is the largest element in t. Thus, the number of subsets for which $f(T) = a$ is finite for all $a \in \mathbb{N}$. Let $U_a = \{T \mid f(T) = a\}$. Clearly, $S = \bigcup_{a=0}^{\infty} U_a$ is the set of all finite subsets of $\mathbb{N}$. We list the elements of S by listing the elements in U_0, the elements in U_1, and so on. The elements in each U_a are listed in some order (e.g. lexicographically). Since every element of S will appear in this list, S is countable.

(c) Say that a function $f : \mathbb{N} \to \{0,1\}$ has *finite support* if it is non-zero on only a finite set of inputs. Let F denote the set of functions $f : \mathbb{N} \to \{0,1\}$ with finite support.

Prove that F is countably infinite.

Answer: We give a bijection between S and $\mathbb{N}$. We encode an $f \in S$ as a binary number y_f, with the ith position (with $i = 0$ being the least significant digit) set to 1 if $f(i) = 1$. Note that this encoding always has finite length, excluding leading zeros, since the maximum i for which $f(i) = 1$ is finite. Thus the encoding always results in a natural number encoded in binary. This conversion is one-to-one, since each f and f' in S differ on at least one input and therefore y_f and $y_{f'}$ differ in at least one position. The conversion is onto, since every binary number represents

a function with finite support. Since the natural numbers are countably infinite, and we have a bijection between S to $\mathbb{N}$, S is countably infinite.

An alternative bijection is between S and the subset of $\mathbb{N}$ that contains only numbers that are the product of distinct primes. Let $\{p_0 = 2, p_1 = 3, p_2 = 5, \cdots\}$ be the set of all primes where p_i is the $(i+1)$th prime. As shown in a previous homework, this set is infinite. Now consider a function $f(x)$ that is 1 exactly on inputs $x_1, x_2, \cdots, x_k$. Encode $f(x)$ as the natural number $p_{x_1} \times p_{x_2} \times \cdots \times p_{x_k}$. In other words, the function f is encoded as the natural number $2^{f(0)} \times 3^{f(1)} \times 5^{f(2)} \times 7^{f(3)} \times 11^{f(4)} \times \cdots$. This encoding is one-to-one, since the prime factorization of a number is unique. The encoding is onto, since every natural number that is composed of distinct primes corresponds to a function in S. Thus this is a bijection, and S is countable.

8. **(2/2/2/2/2) More Countability**

Given:

- A is a countable set, non-empty set. Forall $i \in A$, S_i is an uncountable set.
- B is an uncountable set. Forall $i \in B$, Q_i is a countable set.

For each of the following, decide if the expression is "Always Countable", "Always Uncountable", "Sometimes Countable, Sometimes Uncountable."

For the "Always" cases, prove your claim. For the "Sometimes" case, provide two examples – one where the expression is countable, and one where the expression is uncountable.

(a) $\cup_{i \in A} S_i$

Answer: Always uncountable. Let a be any elem of A. S_a is uncountable. Thus, $\cup_{i \in A} S_i$, a superset of S_a, is uncountable.

(b) $\cap_{i \in A} S_i$

Answer: Sometimes countable, sometimes uncountable.

Countable: When the S_i are disjoint, the intersection is empty, and thus countable. Formally, let $A = N$, let $S_i = \{i\} \times R = \{(i,x) | x \in R\}$. Then, $\cap_{i \in A} S_i = \emptyset$.

Uncountable: When the S_i are identical, the intersection is uncountable. Let $A = N$, let $S_i = R$ forall i. $\cap_{i \in A} S_i = R$ is uncountable.

(c) $\cup_{i \in B} Q_i$

Answer: Sometimes countable, sometimes uncountable.

Countable: Make all th Q_i identical. Formally, let $B = R$, and $Q_i = N$. Then, $\cup_{i \in B} Q_i = N$, is countable.

Uncountable: Let $B = R$. Let $Q_i = \{i\}$. Then, $\cup_{i \in B} Q_i = R$, is uncountable.

(d) $\cap_{i \in B} Q_i$

Answer: Always countable. Let b be any elem of B. Q_b is countable. Thus, $\cap_{i \in B} Q_i$, a subset of Q_b, is also countable.

(e) $A \cap B$

Answer: Always countable. $A \cap B = \emptyset$

CS 70 Discrete Mathematics and Probability Theory
Spring 2016 Rao and Walrand HW 7

Due Thursday March 10th at 10PM

1. **(5 points each, 15 total) Correcting XYZ**

   (a) For any java program P, define $S(P)$ to be the set of all Java programs P' that output the same result as P on all inputs. Formally, $S(P) = \{P' | \forall x : P(x) = P'(x), \text{P'(x) halts if and only if P(x) halts}\}$. XYZ claims that they have built an optimal java-program-shortener. Formally, they claim to have a procedure **optimalShortener** such that:

```
for every java program P:
  let P' = optimalShortener(P)
  then:
    P' is in S(P)
    forall P'' in S(P), length(P'') >= length(P')
```

   Prove that XYZ is wrong.

   Answer:

   We can solve the halting problem with optimalShortener.

```
(using one definition of Halting Problem)
HaltingSolver1(M):

  define M1:
    simulate M(); // suppress any output
    exit 0;

  define M2:
    exit 0;

  if (optimalShortener(M1) == optimalShortener(M2))
    return "halts"
  else
    return "inf loops"



(using other definition of Halting Problem)
HaltingSolver2(M, x):

  define M1:
```

```
    simulate M(x); // suppress any output
    exit 0;

  define M2:
    exit 0;

  if (optimalShortener(M1) == optimalShortener(M2))
    return "halts"
  else
    return "inf loops"
```

(b) Having failed with source code shortening, XYZ now tries their luck with runtime optimization. XYZ claims that they have built a new optimizer **optimizer** such that:

```
for every java program P:
  let P' = optimizer(P)
  then:
    forall x:
      if P(x) halts
      then P'(x) halts within 2^{|x|} steps
      else P'(x) infinite loops
```

Namely, XYZ claims that their optimizer outputs an equivalent program such that: if $P(x)$ halts, then $P'(x)$ will halt within $2^{|x|}$ steps; if $P(x)$ does not halt, then $P'(x)$ does not halt either. Prove that **optimizer** can not exist.

Answer: We can solve the hallting problem with optimizer. To see if $P(x)$ halts, run $P'(x)$ for $2^{|x|}+1$ steps. If it halts, report that $P(x)$ halts. Otherwise, report that $P(x)$ does not halt.

(c) Let XYZ-phone be a smart phone with 16GB of storage, 2GB of RAM, and additional state of 1MB (i.e. CPU registers, etc ..).

Prove that for any program P which does not interact with the rest of the world (no user input, no network connection, no wifi, no bluetooth, no sensors), it is possible to determine whether all executions of P halts or whether some execution of P infinite loops. (Different executions of P may behave differently due to the random number generator, the initial state of the phone when P is loaded, and due to non-determism caused by multi-threading.

Answer: The total state of the machine is less than 17 GB, which is $17 * 2^{32} * 8 < 2^{36}$ bits.

Create a directed graph with $2^{2^{36}}$ nodes, where there is a directed edge from u to v if, in one cpu cycle, the machine can move from state u to state v.

Let S be the set of all nodes reachable by loading porgram P. We the check if there is some node v in S such that v is in some cycle of the graph described above.

2. **(10 points) Printing all x where $M(x)$ halts**

Prove that it is possible to write a program P which:

```
* takes as input M, a java program
* runs forever, and prints out strings to the console
* for every x, if M(x) halts, then P(M) eventually prints out x
* for every x, if M(x) does NOT halt, then P(M) never prints out x
```

Answer:

```
Starting out, let S = {}
for i = 1 to infinity:

  Let N_i = a new machine loaded with program M and input i
  S = S + { N_i }

  simulate every machine in S for 1 cycle

  forall N_x in S that has halted:
    print out x
    remove N_x from S
```

Consider any x, such that $M(x)$ halts after n steps. For some k, at stage k, $M(x)$ is added to S. At state $k+n+1$, $M(x)$ halts, and x is then printed out.

For any x where $M(x)$ does not halt, x is never printed out.

3. **(10 points) Lexicographical output is impossible**

   Lexicographical ordering of strings means (1) shorter strings are in front of longer strings (2) for two strings of the same length, they are sorted in alphabetical order.

   Prove that it's impossible to solve the above problem if we require the output be in lexicographical order.

   Answer:

   Let M be any Java program. We show how to construct a program which can decide whether $M(x)$ halts for any x. Consider the program M2 defined as follows:

```
M2( 2 * k) = simulate M(k)
M2( 2 * k + 1) = halt
```

   Now, suppose there is a lexicographical enumerator for $M2$. Call this enumerator E.

   To decide whether $M(x)$ halts, we run E until we see $2k+1$ printed on the output tape. It must eventually be printed since $M2(2k+1)$ halts.

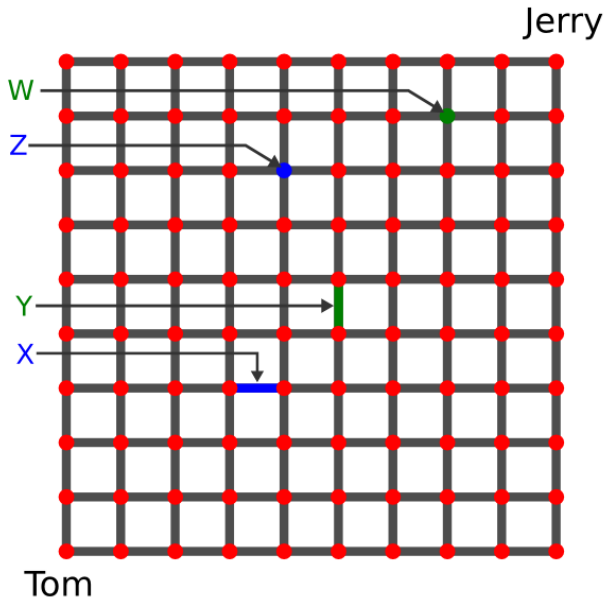
   Then, we see whether $2k$ was printed out the tape already.

   If so, we know that $M2(2k)$ halted, meaing that $M(k)$ halted.

   If not, we know that $M2(2k)$ infinited looped.

4. **(2,2,3,3) Maze**

Let's assume that Tom is located at the bottom left corner of the maze below, and Jerry is located at the top right corner. Tom of course wants to get to Jerry by the shortest path possible.



a) How many such shortest paths exist?

Answer: Each row in the maze has 9 edges, and so does each column. Any shortest path that Tom can take to Jerry will have exactly 9 horizontal edges going right (let's call these "H" edges) and 9 vertical edges going up (let's call these "V" edges).

Observe also that every shortest path from Tom to Jerry can be described by a unique sequence consisting of 9 "H"s and 9 "V"s. For example, one such path is HHHHHHHHHVVVVVVVVV (which represents the path that goes all the way to the right, and then all the way to the top). Conversely, every such sequence of exactly 9 "H"s and 9 "V"s corresponds to a unique shortest path from Tom to Jerry.

Therefore, the number of shortest paths is exactly the same as the number of ways of arranging 9 "H"s and 9 "V"s in a sequence, which is $\binom{18}{9} = 48620$.

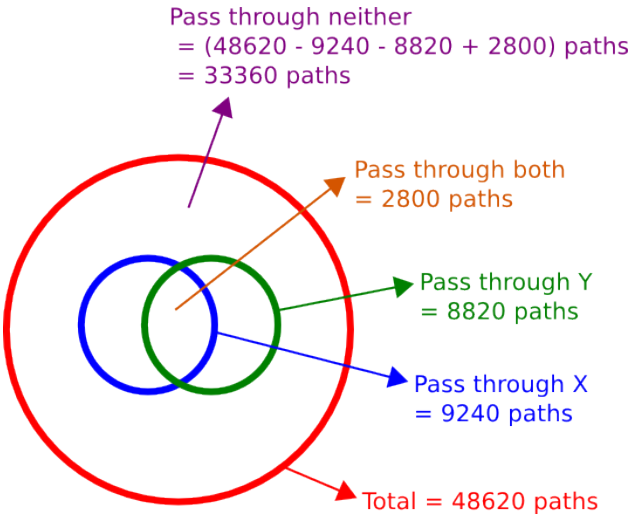
b) How many shortest paths pass through the edge labelled X? The edge labelled Y? Both the edges X and Y? Neither edge X nor edge Y?

Answer: For a shortest path to pass through the edge X, it has to first get to the left vertex of X. So the first portion of the path has to start at the origin, and end at the left vertex of X. Using the same logic as above, there are exactly $\binom{6}{3} = 20$ ways to complete this "first leg" of the path (consisting of 3 "H" edges and 3 "V" edges). Having chosen one of these 20 ways, the path has to then go from the right vertex of X to the top right corner of the maze (the "second leg"). This second leg will consist of 5 "H" edges and 6 "V" edges, and using the same logic, there are exactly $\binom{11}{5} = 462$ possibilities. Therefore, the total number of shortest paths that pass through the edge X is $20 \times 462 = 9240$.

Using similar logic, any shortest path that passes through Y has to consist of 2 legs, the first leg going from the origin to the bottom vertex of Y, and the second leg going from the top vertex of Y

to the top right corner of the maze. The first leg will consist of exactly 5 "H"s and 4 "V"s, while the second leg will consist of exactly 4 "H"s and 4 "V"s. So the total number of such shortest paths will be $\binom{9}{5} \times \binom{8}{4} = 8820$.

By a similar argument, let's try to figure out how many paths will pass through both X and Y. Clearly, any such path has to consist of 3 legs, with the first leg consisting of 3 "H"s and 3 "V"s (going from the origin to the left edge of X), the second leg consisting of 1 "H" and 1 "V" (going from the right vertex of X to the bottom vertex of Y), and the third leg consisting of 4 "H"s and 4 "V"s (going from the top vertex of Y to the top right corner of the maze). The total number of such shortest paths is therefore $\binom{6}{3} \times \binom{2}{1} \times \binom{8}{4} = 2800$.



Finally, we know that there are 48620 shortest paths in all, of which 9240 pass through X, 8820 pass through Y, and 2800 pass through both. So the number of paths that pass through neither is 33360 (see the figure above for an intuitive explanation).

c) How many shortest paths pass through the vertex labelled Z? The vertex labelled W? Both the vertices Z and W? Neither vertex Z nor vertex W?

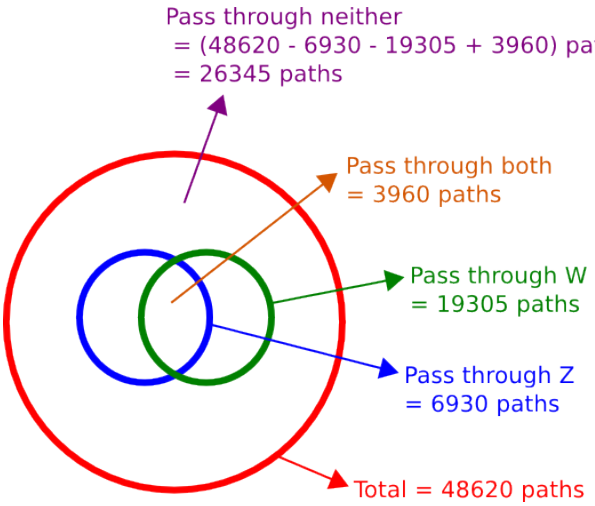
Answer: This part is very similar in spirit to the previous one, except that in this case, each leg of the path we consider begins exactly where the previous leg ended, and *not* to the right or to the top of where the previous leg ended.

For concreteness, let's find out how many shortest paths pass through vertex Z. Observe that for a shortest path to pass through Z, it has to first get to Z. So the first portion of the path has to start at the origin, and end at Z. Using the same logic as above, there are exactly $\binom{11}{4} = 330$ ways to complete this "first leg" of the path (consisting of 4 "H" edges and 7 "V" edges). Having chosen one of these 330 ways, the path has to then go from Z to the top right corner of the maze. This second leg will consist of 5 "H" edges and 2 "V" edges, and so there are exactly $\binom{7}{2} = 21$ possibilities. Therefore, the total number of shortest paths that pass through the vertex Z is $330 \times 21 = 6930$.

Using similar logic, any shortest path that passes through W has to consist of 2 legs, the first leg going from the origin to W, and the second leg going from W to the top right corner of the maze. The first leg will consist of exactly 7 "H"s and 8 "V"s, while the second leg will consist of exactly

2 "H"s and 1 "V". So the total number of such shortest paths will be $\binom{15}{7} \times \binom{3}{1} = 19305$.

By a similar argument, let's try to figure out how many paths will pass through both Z and W. Clearly, any such path has to consist of 3 legs, with the first leg consisting of 4 "H"s and 7 "V"s (going from the origin to Z), the second leg consisting of 3 "H"s and 1 "V" (going from Z to W), and the third leg consisting of 2 "H"s and 1 "V" (going from W to the top right corner of the maze). The total number of such shortest paths is therefore $\binom{11}{4} \times \binom{4}{1} \times \binom{3}{1} = 3960$.



Finally, we know that there are 48620 shortest paths in all, of which 6930 pass through Z, 19305 pass through W, and 3960 pass through both. So the number of paths that pass through neither is 26345 (see the figure above for an intuitive explanation).

5. **(1 point each, 20 total) Counting practice!**
The only way to learn counting is to practice, practice, practice—so here is your chance to do so. No need to justify your answers or show your calculations on this problem. We encourage you to leave your answer as an expression (rather than trying to evaluate it to get a specific number).

   (a) How many 10-bit strings are there that contain exactly 4 ones?
   Answer: We must select 4 of the 10 bits to set to 1. Since we don't care about the order of the selection (i.e. selecting bit 3 before bit 0 is no different from selecting bit 0 before bit 3), the answer is $\binom{10}{4}$. Equivalently, we could choose 6 bits out of the 10 to set to 0. There are $\binom{10}{6}$ ways to do this. The two expressions are the same.

   (b) How many different 13-card bridge hands are there? (A bridge hand is obtained by selecting 13 cards from a standard 52-card deck.' The order of the cards in a bridge hand is irrelevant.)
   Answer: Since the order of the cards in the hand is irrelevant, we again choose 13 of the 52 cards. The answer is $\binom{52}{13}$.

   (c) How many different 13-card bridge hands are there that contain no aces?
   Answer: There are 48 cards that contain no aces and out of these we choose 13, so the answer is $\binom{48}{13}$. In more detail, there are 48 ways to select the first card since we don't want an ace, 47 ways to select the second card, etc. If the order of the choices mattered, we would have $48!/35!$

ways to select the cards, but since order doesn't matter, we divide by 13! and get $\frac{48!}{13!35!} = \binom{48}{13}$.

(d) How many different 13-card bridge hands are there that contain all four aces?
Answer: We know that 4 of the cards in our hand will be aces, so we only have to select the remaining 9. Thus, we choose 9 out of 48 and get the answer $\binom{48}{9}$.

(e) How many different 13-card bridge hands are there that contain exactly 6 spades?
Answer: We first choose the 6 spades from the 13 total spades, then we must choose 7 remaining cards from the 39 non-spades. There are thus $\binom{13}{6}\binom{39}{7}$ ways total.

(f) How many 99-bit strings are there that contain more ones than zeros?
Answer: 2^{98}. There are 2^{99} 99-bit strings total. Since a 99-bit string cannot have equal number of ones and zeroes, by symmetry, there are $2^{99}/2 = 2^{98}$ possible different strings. To see the symmetry, notice that if we have a string of more ones than zeroes, we can flip all the bits and obtain a string of more zeroes than ones and vice versa. Hence, there is a 1-1 correspondence between the strings with more ones and the strings with more zeroes.
Put another way, if S denotes the set of 99-bit strings with more ones than zeros, and T the set of 99-bit strings with more zeros than ones, we see that $S \cap T = \emptyset$ and that $S \cup T =$ the set of all 99-bit strings. By the sum rule, $|S| + |T| = 2^{99}$. Moreover because S can be put into bijective correspondence with T, $|S| = |T|$. Plugging this into the equation above, we see $2 \times |S| = 2^{99}$, so $|S| = 2^{98}$. To put it yet another way, half of all 99-bit strings have more ones than zeros, so the answer is $\frac{1}{2} \times 2^{99} = 2^{98}$.

(g) If we have a standard 52-card deck, how many ways are there to order these 52 cards?
Answer: There are 52 ways to select the topmost card, 51 ways to select the 2nd topmost, etc. There are thus 52! ways total to order the cards.

(h) Two identical decks of 52 cards are mixed together, yielding a stack of 104 cards. How many different ways are there to order this stack of 104 cards?
Answer: First we pretend that the 104 cards are all distinguishable, then there are 104! ways to order them. But we have 52 pairs of identical cards and have counted each configuration twice for each pair. Hence, there are $\frac{104!}{2^{52}}$ possible ways to order the cards.
Here's another way to think about this problem. Out of 104 possible positions, we first choose 2 positions to place the pairs of Ace of Spades; there are $\binom{104}{2}$ ways to do this. Then we have 102 positions left from which we choose 2 positions to place the pairs of Ace of Hearts; there are $\binom{102}{2}$ ways. Using this reasoning, we get the expression $\binom{104}{2}\binom{102}{2}\binom{100}{2}\ldots\binom{2}{2} = \frac{104!}{2^{52}}$.
This is very much like counting the number of anagrams of BONBON, except with 104 letters instead of 6 letters. There are $6!/(2! \cdot 2! \cdot 2!) = 6!/2^3$ anagrams of BONBON. Similarly, there are $8!/2^4$ anagrams of FROUFROU, $10!/2^5$ anagrams of INTESTINES, and ... well, you get the idea.
Note that interestingly enough, if we want to rephrase this problem in term of balls and bins, we would let the position be the balls and let the card face be the bins even though we intuitively think we should be placing the cards into a position. So, this problem is really the same as finding the number of ways of throwing 104 labelled balls into 52 bins such that every bin has exactly 2 balls.

(i) How many different anagrams of FLORIDA are there? (An anagram of FLORIDA is any reordering of the letters of FLORIDA, i.e., any string made up of the letters F, L, O, R, I, D, and A, in any order. The anagram does not have to be an English word.)
Answer: Since the order of the letters matter and all of the letters are distinct, there are 7! different anagrams.

(j) How many different anagrams of ALASKA are there?
Answer: If we first pretend that the 3 A's are all distinct (i.e. subscripted), then there are 6! anagrams. But since the 3 A's are identical, we counted each anagram an extra 3! ways. Hence, there are $6!/3!$ anagrams total. Another way to think about this: we first choose 3 of out of the 6 possible positions to place the A, there are $\binom{6}{3}$ choices. There are then 3 positions left to place the L, 2 positions to place the S, and one position to place the A, so there are $\binom{6}{3}(3!) = 6!/3!$ anagrams total.

(k) How many different anagrams of ALABAMA are there?
Answer: Similar to previous problem, there are $7!/4!$ anagrams total.

(l) How many different anagrams of MONTANA are there?
Answer: If we pretend the N's are distinct, then there would be $7!/2!$ anagrams. But since the N's are identical, we counted each configuration twice and must divide by an additional 2!. So our final answer is $\frac{7!}{2!2!}$.

(m) We have 9 balls, numbered 1 through 9, and 27 bins. How many different ways are there to distribute these 9 balls among the 27 bins?
Answer: Each ball can go into any one of the 27 bins. So there are 27^9 possible ways. One can also view this problem as asking for the number of functions that map the balls into the bins.

(n) We throw 9 identical balls into 7 bins. How many different ways are there to distribute these 9 balls among the 7 bins such that no bin is empty?
Answer: The answer is the same as the number of ways to distribute 2 balls among 7 bins. Once we've distributed those 2 balls in any way whatsoever, then we can add one ball to each bin, yielding a configuration with 9 balls in 7 bins such that no bin is empty. There is bijective correspondence between ways to distribute 2 balls among 7 bins and ways to distribute 9 balls among 2 bins such that no bin is empty. There are 7 ways to distribute 2 balls into the bins so that both balls fall into the same bin, and $\binom{7}{2}$ ways to distribute 2 balls into the bins so that both balls fall into different bins, so the total number of ways is $7 + \binom{7}{2}$.
Alternate solution: This is a stars and bars problem. The bars represent the dividers between the bins, and each star represents one ball. We require there to be at least one star between every bar and the leftmost and rightmost character cannot be a bar. Consequently we are actually counting the number of ways to create a binary string of length $6+9 = 15$ where there are 6 one-bits and no two one-bits are adjacent. We can use problem 2 to get $\binom{8}{2} = \binom{8}{6} = \binom{7}{5} + \binom{7}{6}$ as our final answer.
Alternate solution: In every configuration, there can either be 5 bins with 1 ball in each and 2 bins with 2 balls in each, or there could be 6 bins with 1 ball and 1 bin with 3 balls. In the first case, there are $\binom{7}{5}$ ways to choose the 5 1-ball bins and once the 1-ball bins are chosen,

we have no more choices for the 2-ball bins so we are done. In the second case, there are $\binom{7}{6}$ ways to choose the 6 1-ball bins and hence the total ways to distribute the 9 balls is just $\binom{7}{5} + \binom{7}{6}$.

(o) How many different ways are there to throw 9 identical balls into 27 bins?
Answer: This is a stars and bars problem. The bars represent dividers between the bins, and each star represents one ball. We wish to insert 26 bars in between the 9 stars, which corresponds to choosing a 35-bit string that has exactly 9 zero-bits. There are thus $\binom{35}{9}$ ways to distribute the balls. You should contrast this with the distinguishable balls case in part (m).

(p) There are exactly 132 students currently enrolled in CS70. How many different ways are there to pair up the 132 CS70 students, so that each student is paired with one other student?
Answer: This problem is equivalent to throwing 132 balls into 66 **identical** bins such that every bin has two balls. We first imagine that the bins are not identical; then there would be $\frac{132!}{2^{66}}$ ways to throw the balls as shown before. But since we can't distinguish between the case where ball A and B are in bin 1 and ball C is in bin 4 vs. the case where ball A and B are in bin 4 and ball C is in bin 1, we can permute the content of the bins and still get the same configuration. Thus we know that we over-counted by a factor of 66! and derive $\frac{132!}{2^{66}66!}$ as our final answer.
Alternative solution: Another way to approach this problem is to first send 66 students to the moon. The selection order doesn't matter in this first choice so there are $\binom{132}{66}$ ways to do this. Then we match the remaining 66 students on Earth one by one with the students on the moon. There are 66! ways to perform the matching. But in this scheme, we counted sending student Andy to the moon and matching him with student Betty on Earth as different configuration from sending student Betty to the moon and matching her with student Andy on Earth. Since we double counted for each pair of students, we divide by 2^{66} to get $(\binom{132}{66}66!)/2^{66} = \frac{132!}{2^{66}66!}$ as our final expression.
Alternative solution: Here's a naive approach. We have 132 students in the classroom. Pick any two of them ($\binom{132}{2}$ ways to do this), pair them up, and send them home. Now there are 130 students. Pick two of them ($\binom{130}{2}$ ways), pair 'em, and send 'em home. Repeat until there's no one left anymore. In total there are $\binom{132}{2}\binom{130}{2}\binom{128}{2}\cdots\binom{2}{2}$ ways to do this.
But wait! This overcounts shamelessly. Suppose we had four students. Picking Alice and Betty first followed by picking Carol and Dave would be equivalent to first picking Carol and Dave followed by picking Alice and Betty. So, with four students, we'd be overcounting by a factor of two. With $2n$ students, we're overcounting by a factor of $n!$, since there are $n!$ different orders in which we could have chosen the n pairs and they all lead to the same pairing. Consequently in the original problem we've overcounted by a factor of 66!. So the final answer is $\binom{132}{2}\binom{130}{2}\binom{128}{2}\cdots\binom{2}{2}/66!$ ways to do this.

(q) How many ways are there to arrange n 1s and k 0s into a sequence?

Answer: $\binom{n+k}{k}$

(r) How many solutions does

$x_0 + x_1 + \ldots + x_k = n$

have, if all xs must be non-negative integers?

Answer: $\binom{n+k}{k}$. There is a bijection between a sequence of n ones and k plusses and a solution to the equation: x_0 is the number of ones before the first plus, x_1 is the number of ones between the first and second plus, etc. A key idea is that if a bijection exists between two sets they must be the same size, so counting the elements of one tells us how many the other has.

(s) How many solutions does

$x_0 + x_1 = n$

have, if all xs must be *strictly positive* integers?

Answer: $n-1$. It's easy just to enumerate the solutions here. x_0 can take values $1, 2, \ldots, n-1$ and this uniquely fixes the value of x_1. So, we have $n-1$ ways to do this. But, this is just an example of the more general question below.

(t) How many solutions does

$x_0 + x_1 + \ldots + x_k = n$

have, if all xs must be *strictly positive* integers?

Answer: $\binom{(n-(k+1))+k}{k} = \binom{n-1}{k}$. By subtracting 1 from all $k+1$ variables, and $k+1$ from the total required, we reduce it to problem with the same form as the previous problem. Once we have a solution to that we reverse the process, and adding 1 to all the non-negative variables gives us positive variables.

6. **(2 points each, 10 total) Prove the following identities by combinatorial argument:**

(a)

$$\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$$

Answer: The left hand side is the number of ways to choose k elements out of n. Looking at this another way, we look at the first element and decide whether we are going to choose it or not. If we choose it, then we need to choose $k-1$ more elements from the remaining $n-1$. If we don't choose it, then we need to choose all our k elements from the remaining $n-1$. We are not double counting, since in one of our cases we chose the first element and in the other, we did not.

(b)

$$\binom{2n}{2} = 2\binom{n}{2} + n^2.$$

Answer: The left hand side is the number of ways to choose two elements out of $2n$. Counting in another way, we first divide the $2n$ elements (arbitrarily) into two sets of n elements. Then we consider three cases: either we choose both elements out of the first n-element set, both out of the second n-element set, or one element out of each set. The number of ways we can do each of these things is $\binom{n}{2}$, $\binom{n}{2}$, and n^2, respectively. Since these three cases are mutually exclusive and cover all the possibilities, summing them must give the same number as the left hand side. This completes the proof.

(c)

$$\sum_{k=0}^{n} k\binom{n}{k} = n2^{n-1}$$

Answer: RHS: From n people, pick one team-leader and some (possibly empty) subset of other people on his team.

LHS: First pick k people on the team, then pick the leader among them.

(d)

$$\sum_{k=j}^{n} \binom{n}{k}\binom{k}{j} = 2^{n-j}\binom{n}{j}$$

Answer: RHS: Form a team as follows: Pick j leaders from n people. Then pick some (possibly empty) subset of the remaining people.

LHS: First pick $k \geq j$ people on the team, then pick the j leaders among them.

(e)

$$\sum_{i=0}^{k} \binom{m}{i}\binom{n}{k-i} = \binom{m+n}{k}$$

Answer: We'll start with the right-hand side. This is counting the number of ways to pick k things from $m+n$ objects. The left-hand side is summing up, for all possible values of i, the ways to pick i things from m and $k-i$ things from n. We see that the two are equivalent.

CS 70 Discrete Mathematics and Probability Theory
Spring 2016 Rao and Walrand HW 8

Due Thursday March 17th at 10PM

1. **Independence**

   2 points per sub problem. 16 points total.

   (a) **Independence (due to H.W. Lenstra)**

   Suppose we pick a random card from a standard deck of 52 playing cards. Let A represent the event that the card is an queen, B the event that the card is a spade, and C the event that a red card (a heart or a diamond) is drawn.

   i. Which two of A, B, and C are independent? Justify your answer carefully. (In other words: For each pair of events (AB, AC, and BC), state and prove whether they are independent or not.)

   Answer: A and B are independent, since $\Pr[A \cap B] = 1/52 = 1/13 \times 1/4 = \Pr[A]\Pr[B]$.
   A and C are independent, since $\Pr[A \cap C] = 2/52 = 1/13 \times 2/4 = \Pr[A]\Pr[C]$.
   B and C are *not* independent, since $\Pr[B \cap C] = 0 \neq 1/4 \times 2/4 = \Pr[B]\Pr[C]$.

   ii. What if a joker is added to the deck? Justify your answer carefully.

   Answer: Let A', B', C' denote the corresponding events when a joker is added to the deck. I assume that the joker has neither suit, rank, nor color, so that the joker is neither a queen, a spade, nor a red card. Then:
   A' and B' are *not* independent, since $\Pr[A' \cap B'] = 1/53 \neq 4/53 \times 13/53 = \Pr[A']\Pr[B']$.
   A' and C' are *not* independent, since $\Pr[A' \cap C'] = 2/53 \neq 4/53 \times 26/53 = \Pr[A']\Pr[C']$.
   B' and C' are *not* independent, since $\Pr[B' \cap C'] = 0 \neq 13/53 \times 26/53 = \Pr[B']\Pr[C']$.

   (b) **Independence (due to H.W. Lenstra)**

   Let Ω be a sample space, and let $A, B \subseteq \Omega$ be two *independent* events. Let $\overline{A} = \Omega - A$ and $\overline{B} = \Omega - B$ (sometimes written $\neg A$ and $\neg B$) denote the complementary events.

   For the purposes of this question, you may use the following definition of independence: Two events A, B are *independent* if $\Pr[A \cap B] = \Pr[A]\Pr[B]$.

   i. Prove or disprove: $\overline{A}$ and $\overline{B}$ are necessarily independent.

Answer: True. $\overline{A}$ and $\overline{B}$ must be independent:

$$\begin{aligned}
\Pr[\overline{A} \cap \overline{B}] &= \Pr[\overline{A \cup B}] && \text{(by De Morgan's law)} \\
&= 1 - \Pr[A \cup B] && \text{(since } \Pr[\overline{E}] = 1 - \Pr[E] \text{ for all } E\text{)} \\
&= 1 - (\Pr[A] + \Pr[B] - \Pr[A \cap B]) && \text{(union of overlapping events)} \\
&= 1 - \Pr[A] - \Pr[B] + \Pr[A]\Pr[B]) \\
& && \text{(using our assumption that } A \text{ and } B \text{ are independent)} \\
&= (1 - \Pr[A])(1 - \Pr[B]) \\
&= \Pr[\overline{A}]\Pr[\overline{B}] && \text{(since } \Pr[\overline{E}] = 1 - \Pr[E] \text{ for all } E\text{)}
\end{aligned}$$

ii. Prove or disprove: A and $\overline{B}$ are necessarily independent.

Answer: True. A and $\overline{B}$ must be independent:

$$\begin{aligned}
\Pr[A \cap \overline{B}] &= \Pr[A - (A \cap B)] \\
&= \Pr[A] - \Pr[A \cap B] \\
&= \Pr[A] - \Pr[A]\Pr[B] \\
&= \Pr[A](1 - \Pr[B]) \\
&= \Pr[A]\Pr[\overline{B}]
\end{aligned}$$

iii. Prove or disprove: A and $\overline{A}$ are necessarily independent.

Answer: False in general. If $0 < \Pr[A] < 1$, then $\Pr[A \cap \overline{A}] = \Pr[\emptyset] = 0$ but $\Pr[A]\Pr[\overline{A}] > 0$, so $\Pr[A \cap \overline{A}] \neq \Pr[A]\Pr[\overline{A}]$; therefore A and $\overline{A}$ are not independent in this case.

iv. Prove or disprove: It is possible that $A = B$.

Answer: True. To give one example, if $\Pr[A] = \Pr[B] = 0$, then $\Pr[A \cap B] = 0 = 0 \times 0 = \Pr[A]\Pr[B]$, so A and B are independent in this case. (Another example: If $A = B$ and $\Pr[A] = 1$, then A and B are independent.)

(c) **Bonferroni's inequalities**

i. For events A, B in the same probability space, prove that

$$\Pr[A \cap B] \geq \Pr[A] + \Pr[B] - 1.$$

ii. Generalize part (a) to prove that, for events $A_1, \ldots, A_n$ in the same probability space (and any n),

$$\Pr[A_1 \cap \ldots \cap A_n] \geq \Pr[A_1] + \cdots + \Pr[A_n] - (n-1).$$

Answer:

i. To show this we use the Inclusion-Exclusion theorem. We have that for all events A and B,

$$\Pr[A \cup B] = \Pr[A] + \Pr[B] - \Pr[A \cap B].$$

Also, we know that the probability of any event is at most 1. Thus, $\Pr[A \cup B] \leq 1$. Using this with the Inclusion-Exclusion theorem, we get

$$\Pr[A \cap B] = \Pr[A] + \Pr[B] - \Pr[A \cup B] \geq \Pr[A] + \Pr[B] - 1.$$

ii. Now we generalize this result to show that:

$$\Pr[A_1 \cap A_2 \cap \cdots \cap A_n] \geq \Pr[A_1] + \Pr[A_2] + \cdots + \Pr[A_n] - (n-1). \quad (1)$$

For this, we use induction on n.
The base case $n = 1$ just says that $\Pr[A_1] \geq \Pr[A_1] - 0$, which is trivially true.
For our inductive hypothesis, we assume that equation (1) holds for some arbitrary n and any n events.
The inductive step, therefore, is to show that it holds for $n+1$. In other words we need to show:

$$\Pr[A_1 \cap A_2 \cap \cdots \cap A_n \cap A_{n+1}] \geq \Pr[A_1] + \Pr[A_2] + \cdots + \Pr[A_n] + \Pr[A_{n+1}] - n. \quad (2)$$

Now, let B denote the event $A_n \cap A_{n+1}$. Then we have, by the inductive hypothesis applied to the n events $A_1, A_2, \ldots, A_{n-1}, B$,

$$\Pr[A_1 \cap A_2 \cap \cdots \cap A_{n-1} \cap B] \geq \Pr[A_1] + \Pr[A_2] + \cdots + \Pr[A_{n-1}] + \Pr[B] - (n-1). \quad (3)$$

However, from our proof for the case $n = 2$ in part (a) we have:

$$\Pr[B] = \Pr[A_n \cap A_{n+1}] \geq \Pr[A_n] + \Pr[A_{n+1}] - 1.$$

Substituting this into equation (3) gives us our desired equation (2), which completes the induction proof.

2. **(1/2/2) Cliques in random graphs**

Consider a graph $G(V,E)$ on n vertices which is generated by the following random process: for each pair of vertices u and v, we flip a fair coin and place an (undirected) edge between u and v if and only if the coin comes up heads. So for example if $n = 2$, then with probability $1/2$, $G(V,E)$ is the graph consisting of two vertices connected by an edge, and with probability $1/2$ it is the graph consisting of two ianswerated vertices.

(a) What is the size of the sample space?

(b) A k-clique in graph is a set of k vertices which are pairwise adjacent (every pair of vertices is connected by an edge). For example a 3-clique is a triangle. What is the probability that a particular set of k vertices forms a k-clique?

(c) Prove that the probability that the graph contains a k-clique for $k = 4\lceil \log n \rceil + 1$ is at most $1/n$.

Answer:

(a) There are two choices for each of the $\binom{n}{2}$ pairs of vertices, so the size of the sample space is $2^{\binom{n}{2}}$.

(b) For a xed set of k vertices to be a k-clique, all of the $\binom{k}{2}$ pairs of those vertices have to be connected by an edge. The probability of this event is $1/2^{\binom{k}{2}}$.

(c) Let A_S denote the event that S is a k-clique, where $S \subseteq V$ is of size k. Then, the event that the graph contains a k-clique can be described as the union of A_S's over all $S \subseteq V$ of size k. Using the union bound,

$$\Pr\left[\bigcup_{S \subseteq V, |S|=k} A_S\right] \leq \sum_{S \subseteq V, |S|=k} \Pr[A_S] = \sum_{S \subseteq V, |S|=k} \frac{1}{2^{\binom{k}{2}}}.$$

Now, since there are $\binom{n}{k}$ ways of choosing a subset $S \subseteq V$ of size k, the right-hand side of the above equality is

$$\frac{\binom{n}{k}}{2^{\binom{k}{2}}} = \frac{\binom{n}{k}}{2^{\frac{k(k-1)}{2}}} \leq \frac{n^k}{\left(2^{\frac{(k-1)}{2}}\right)^k} \leq \frac{n^k}{\left(2^{\frac{(4\log n+1-1)}{2}}\right)^k} = \frac{n^k}{(2^{2\log n})^k} = \frac{n^k}{n^{2k}} = \frac{1}{n^k} \leq \frac{1}{n}.$$

3. **(1/2/2) College applications**

There are n students applying to n colleges. Each college has a ranking over all students (i.e. a permutation) which, for all we know, is completely random and independent of other colleges.

College number i will admit the first k_i students in its ranking. If a student is not admitted to any college, he or she might le a complaint against the board of colleges, and colleges want to avoid that as much as possible.

a) If for all i, $k_i = 1$, i.e. if every college only admits the top student on its list, what is the chance that all students will be admitted to at least one college?

Answer: If we consider the first choices of all colleges, there are n^n different possibilities, all of which are equally likely because colleges are indepently sorting students in a random manner. Out of these we want the possibilities that have all students covered, which is the same as those that have no repeated student (because the number of colleges is the same as the number of students). So we are counting permutations, and we know that there are $n!$ of them. So the probability is $\frac{n!}{n^n}$.

b) What is the chance that a particular student, Alice, does not get admitted to any college? Prove that if the average of all k_i's is $2\ln n$, then this probability is at most $1/n^2$. (Hint: use the inequality $1-x < e^{-x}$)

Answer: The chance that Alice does not get admitted to college i is $1-\frac{k_i}{n}$. This is because out of all the $n!$ permutations that college i can have on students $k_i \times (n-1)!$ of them result in Alice being one of the top k_i (we first choose Alice's place and then randomly permute the remaining students). So the probability that Alice ends up in the top k_i is k_i/n and the probability that she does not is $1-\frac{k_i}{n}$.
The probability that she does not get admitted to any college is just

$$\prod_{i=1}^{n}(1-\frac{k_i}{n})$$

Now using the inequality $1-x \leq e^{-x}$, we get $1-\frac{k_i}{n} \leq e^{-k_i/n}$. Multiplying over all i we get

$$\prod_{i=1}^{n}(1-\frac{k_i}{n}) < \prod_{i=1}^{n} e^{-k_i/n} = e^{-\Sigma_{i=1}^{n} k_i/n}$$

But $\sum_{i=1}^{n} k_i/n$ is simply the average of all k_i. If this average is $2\ln n$, the last expression simply reduces to $e^{-2\ln n}$ which is just $1/n^2$.

c) Prove that when the average k_i is $2\ln n$, then the probability that at least one student does not get admitted to any college is at most $1/n$. (Hint: use the union bound)

Answer: If A_i is the event that student i does not get admitted to any college is at most $1/n^2$ by the previous part. $\cup_{i=1}^{n} A_i$ is the event that at least one of the students does not get admitted to any college.

By using the union bound we get

$$\Pr[\cup_{i=1}^{n} A_i] \leq \sum_{i=1}^{n} \Pr[A_i] \leq \sum_{i=1}^{n} \frac{1}{n^2} = \frac{1}{n}.$$

4. **Expressions**

- Each subpart is 1 point. 21 points total.
- For each problem, just write down a mathematical expression. There is no need to justify/explain/derive the answer.

(a) **Bayes Rule - Man Speaks Truth**

i. A man speaks the truth 3 out of 4 times. He flips a biased coin that comes up Heads $\frac{1}{3}$ of the time and reports it's Heads. What is the probability it is Heads?

ii. A man speaks the truth 3 out of 4 times. He rolls a fair 6-sided dice and reports it comes up 6. What is the probability it is really 6?

Answer:

i. Let E denotes the event the man reports heads, S_1 be the event that the coin comes up heads, and S_2 be the event that the coin comes up tails.
We have: $P(E|S_1) = \frac{3}{4}, P(E|S_2) = \frac{1}{4}, P(S_1) = \frac{1}{3}, P(S_2) = \frac{2}{3}$.
We want to compute $P(S_1|E)$, and let's do so by applying Bayes Rule.
$P(S_1|E) = \frac{P(S_1E)}{P(E)} = \frac{P(E|S_1)P(S_1)}{P(E|S_1)P(S_1)+P(E|S_2)P(S_2)} = \frac{3/4 \cdot 1/3}{3/4 \cdot 1/3 + 1/4 \cdot 2/3} = \frac{3}{5}$.

ii. Let D be the event that the dice rolls a 6. Let M be the event that the man says 6.
$P(D|M) = \frac{P(D \wedge M)}{P(M)} = \frac{P(M|D)P(D)}{P(M|D)P(D) + P(M|\neg D)P(\neg D)} = \frac{3/4 * 1/6}{3/4 * 1/6 + 1/4 * 1/5 * 5/6} = \frac{3/24}{4/24} = \frac{3}{4}$

(b) **Unlikely events**

i. Toss a fair coin x times. What is the probability that you never get heads?
Answer: 0.5^x

ii. Roll a fair die x times. What is the probability that you never roll a six?
Answer: $(1 - \frac{1}{6})^x$

iii. Suppose your weekly local lottery has a winning chance of $1/10^6$. You buy lottery from them for x weeks in a row. What is the probability that you never win?
Answer: $(1 - 1/10^6)^x$

iv. How large must x be so that you get a head with probability at least 0.9? Roll a 6 with probability at least 0.9? Win the lottery with probability at least 0.9?
Answer: For coin, want: $0.5^x \leq 0.1$ so $x \geq \frac{\log 0.1}{\log 0.5} \approx 3.32$

For die, want: $(5/6)^x \leq 0.1$ so $x \geq \frac{\log 0.1}{\log 5/6} \approx 12.6$

For coin, want: $(1-1/10^6)^x \leq 0.1$ so $x \geq \frac{\log 0.1}{\log 1-1/10^6} \approx 2*10^6$ Comment on how answer for coin is almost exactly equal to $\log 0.1/(1/10^6)$ using the approximation $(1-x) \approx e^{(-x)}, xbeing 1/10^6$

(c) **Blood Type**

Consider the three alleles, A, B, and O, for human blood types. As each person inherits one of the 3 alleles from each parent, there are 6 possible genotypes: AA, AB, AO, BB, BO, and OO. Blood groups A and B are dominant to O. Therefore, people with AA or AO have type A blood. Similarly, BB and BO result in type B blood. The AB genotype is called type AB blood, and the OO genotype is called type O blood. Each parent contributes one allele randomly. Now, suppose that the frequencies of the A, B, and O alleles are 0.4, 0.25, and 0.35, respectively, in Berkeley. Alice and Bob, two residents of Berkeley are married and have a daughter, Mary. Alice has blood type AB.

i. What is the probability that Bob's genotype is AO?

Answer: Let B_{1A}, B_{1B} and B_{1O} be the events that Bob's first allele is A, B, and O, respectively. Let B_{2A}, B_{2B} and B_{2O} be the events that Bob's second allele is A, B, and O respectively. Bob's blood type can be AA, AB, AO, BB, or BO. Let B_{AA} be the event that Bob has type AA blood, B_{AB} be the event that Bob has type AB blood, B_{AO} be the event that Bob has type AO blood, B_{BB} be the event that Bob has type BB blood, and B_{BO} be the event that Bob has type BO blood. The sample space is $\Omega = \{B_{AA}, B_{AB}, B_{AO}, B_{BB}, B_{BO}\}$. Note that we have

$$
\begin{aligned}
B_{AA} &= B_{1A} \cap B_{2A} \\
B_{AB} &= (B_{1A} \cap B_{2B}) \cup (B_{1B} \cap B_{2A}) \\
B_{AO} &= (B_{1A} \cap B_{2O}) \cup (B_{1O} \cap B_{2A}) \\
B_{BB} &= B_{1B} \cap B_{2B} \\
B_{BO} &= (B_{1B} \cap B_{2O}) \cup (B_{1O} \cap B_{2B}).
\end{aligned}
$$

Since the first allele and second allele don't know about each other, the occurrence of the first allele will not affect the second, and vice versa. Therefore, using the rules that $P(A \cap B) = P(A|B)P(B)$ and $P(A \cup B) = P(A) + P(B) - P(A \cap B)$,

$$
\begin{aligned}
P(B_{AA}) &= P(B_{1A}|B_{2A})P(B_{2A}) = P(B_{1A})P(B_{2A}) = (.4)(.4) = .16 \\
P(B_{AB}) &= P(B_{1A}|B_{2B})P(B_{2B}) + P(B_{1B}|B_{2A})P(B_{2A}) - P((B_{1A} \cap B_{2B}) \cap (B_{1B} \cap B_{2A})) \\
&= P(B_{1A})P(B_{2B}) + P(B_{1B})P(B_{2A}) = 2(.4)(.25) = .2 \\
P(B_{AO}) &= P(B_{1A}|B_{2O})P(B_{2O}) + P(B_{1O}|B_{2A})P(B_{2A}) - P((B_{1A} \cap B_{2O}) \cap (B_{1O} \cap B_{2A})) \\
&= P(B_{1A})P(B_{2O}) + P(B_{1O})P(B_{2A}) = 2(.4)(.35) = .28 \\
P(B_{BB}) &= P(B_{1B}|B_{2B})P(B_{2B}) = P(B_{1B})P(B_{2B}) = (.25)(.25) = .0625 \\
P(B_{BO}) &= P(B_{1B}|B_{2O})P(B_{2O}) + P(B_{1O}|B_{2B})P(B_{2B}) - P((B_{1B} \cap B_{2O}) \cap (B_{1O} \cap B_{2B})) \\
&= P(B_{1B})P(B_{2O}) + P(B_{1O})P(B_{2B}) = 2(.25)(.35) = .175.
\end{aligned}
$$

Therefore $P(B_{AO}) = .28$.

ii. Assume that Bob's genotype is AO. What is the probability that Mary's blood type is AB?

Answer: Since Alice has type AB and Bob has type AO, the sample space of possible genotypes for Mary is $\{AA, AO, AB, BO\}$. Since there is uniform probability of inheriting either allele from a given parent, there is a $1/4$ chance that Mary will have type AB blood.

iii. Assume Mary's blood type is AB. What is the probability that Bob's genotype is AA?

Answer: Bob's blood type can be AA, AB, AO, BB, or BO. As in part (a), let B_{AA} be the event that Bob has type AA blood, B_{AB} be the event that Bob has type AB blood, B_{AO} be the event that Bob has type AO blood, B_{BB} be the event that Bob has type BB blood, and B_{BO} be the event that Bob has type BO blood. We already computed the probability of Bob having these blood types in part (a):

$$\begin{aligned}
Pr(B_{AA}) &= .16 \\
Pr(B_{AB}) &= .2 \\
Pr(B_{AO}) &= .28 \\
Pr(B_{BB}) &= .0625 \\
Pr(B_{BO}) &= .175.
\end{aligned}$$

Now, let the event that Mary has blood type AB be M_{AB}. The problem asks us to nd $Pr(B_{AA}|M_{AB})$. We can compute this using Bayes' formula, which says that

$$Pr(B_{AA}|M_{AB}) = \frac{Pr(M_{AB}|B_{AA}) \cdot Pr(B_{AA})}{Pr(M_{AB})}.$$

To nd $Pr(M_{AB})$, we can use the Law of Total Probability, which says that

$$\begin{aligned}
Pr(M_{AB}) = {} & Pr(M_{AB}|B_{AA}) \cdot Pr(B_{AA}) + Pr(M_{AB}|B_{AB}) \cdot Pr(B_{AB}) + Pr(M_{AB}|B_{AO}) \cdot Pr(B_{AO}) \\
& + Pr(M_{AB}|B_{BB}) \cdot Pr(B_{BB}) + Pr(M_{AB}|B_{BO}) \cdot Pr(B_{BO}).
\end{aligned}$$

To calculate this, we must nd the conditional probabilities that Mary has AB blood given Bob's blood type. Recall that Alice has type AB blood.

- If Bob has AA blood, the possible combinations of their alleles are AA, AA, AB, and AB, so $Pr(M_{AB}|B_{AA}) = 1/2$.
- If Bob has AB blood, the possible combinations of their alleles are AA, AB, AB, and BB, so $Pr(M_{AB}|B_{AB}) = 1/2$.
- If Bob has AO blood, the possible combinations of their alleles are AA, AO, AB, and BO, so $Pr(M_{AB}|B_{AO}) = 1/4$.
- If Bob has BB blood, the possible combinations of their alleles are AB, AB, BB, and BB, so $Pr(M_{AB}|B_{BB}) = 1/2$.
- If Bob has BO blood, the possible combinations of their alleles are AB, AO, BB, and BO, so $Pr(M_{AB}|B_{BO}) = 1/4$.

We now have all the information we need to plug in and answerve. By the Law of Total Probability above, we have

$$\begin{aligned}
Pr(M_{AB}) &= Pr(M_{AB}|B_{AA}) \cdot Pr(B_{AA}) + Pr(M_{AB}|B_{AB}) \cdot Pr(B_{AB}) + Pr(M_{AB}|B_{AO}) \cdot Pr(B_{AO}) \\
&\quad + Pr(M_{AB}|B_{BB}) \cdot Pr(B_{BB}) + Pr(M_{AB}|B_{BO}) \cdot Pr(B_{BO}) \\
&= (.5)(.16) + (.5)(.2) + (.25)(.28) + (.5)(.0625) + (.25)(.175) \\
&= .325,
\end{aligned}$$

and plugging in to Bayes' formula, we nd that

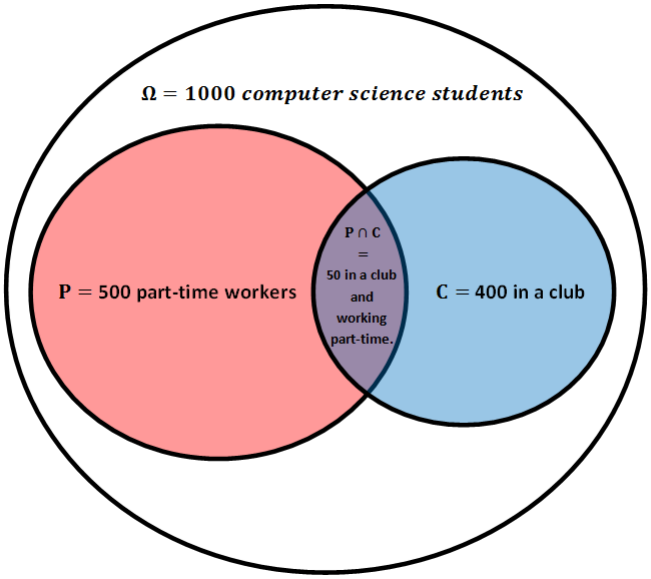
$$Pr(B_{AA}|M_{AB}) = \frac{Pr(M_{AB}|B_{AA}) \cdot Pr(B_{AA})}{Pr(M_{AB})} = \frac{(.5)(.16)}{.325} = .246.$$

(d) **Simple probability**

Out of 1000 sophomore EECS students, 400 are taking CS70 (and may concurrently take CS61C), 500 are taking CS61C (and may concurrently take CS70), and 50 are taking both CS70 and CS61C.

i. Suppose we choose a student uniformly at random. Let C be the event that the student takes CS70 and P the event that the student takes CS61C. Draw a picture of the sample space Ω and the events C and P.

Answer: The following is the sample space. (Belong to a club = taking CS70, work part time = taking CS61C)



ii. What is the probability that the student takes CS70?

Answer: $\Pr[C] = \frac{|C|}{|\Omega|} = \frac{400}{1000} = \boxed{.4}$

iii. What is the probability that the student takes CS61C?

Answer: $\Pr[P] = \frac{|P|}{|\Omega|} = \frac{500}{1000} = \boxed{.5}$

iv. What is the probability that the student takes CS70 AND CS61C?

Answer: $\Pr[P \cap C] = \frac{|P \cap C|}{|\Omega|} = \frac{50}{1000} = \boxed{.05}$

v. What is the probability that the student takes CS70 OR CS61C?

Answer: $\Pr[P \cup C] = \Pr[P] + \Pr[C] - \Pr[P \cap C] = \boxed{.85}$. This answer also comes from something called the Inclusion-exclusion principle.

(e) **Roll Dice**

You roll three fair six-sided dice. What is the probability of rolling a triple (all three dice agree)? What is the probability of rolling a double (two of the dice agree with each other)?

Answer: The sample space Ω consists of all possible outcomes of rolling 3 dies. Therefore, the size of it is: $|\Omega| = 6^3$. Let A be the event of rolling a triple, B be the event of rolling a double. The size of A is 6 since A consists of three ones', three twos', etc.

$$P[\text{rolling a triple}] = \frac{|A|}{|\Omega|} = \frac{6}{6^3}$$

The size of B is $6 \cdot 5 \cdot \frac{3!}{2!1!}$ because you have 6 ways to choose a number that appears twice in the roll, 5 ways to choose a number that is different the previous number and appears once in the

roll. And you have $\frac{3!}{2!1!}$ possible different arrangement for two identical number and one distinct number. So we have

$$P[\text{rolling a double}] = \frac{|B|}{|\Omega|} = \frac{6 \cdot 5 \cdot 3}{6^3} = \frac{5}{12}.$$

(f) **Lie Detector**

A lie detector is known to be 80% reliable when the person is guilty and 95% reliable when the person is innocent. If a suspect is chosen from a group of suspects of which only 1% have ever committed a crime, and the test indicates that the person is guilty, what is the probability that he is innocent?

Answer: Let A denote the event that the test indicates that the person is guilty, and B the event that the person is innocent. Note that

$$\Pr[B] = 0.99, \quad \Pr[\overline{B}] = 0.01, \quad \Pr[A \mid B] = 0.05, \quad \Pr[A \mid \overline{B}] = 0.8$$

Using the Bayesian Inference Rule, we can compute the desired probability as follows:

$$\Pr[B \mid A] = \frac{\Pr[B]\Pr[A \mid B]}{\Pr[B]\Pr[A \mid B] + \Pr[\overline{B}]\Pr[A \mid \overline{B}]} = \frac{0.99 \cdot 0.05}{0.99 \cdot 0.05 + 0.01 \cdot 0.8} \approx 0.86$$

(g) **Rain and Wind**

The local weather channel just released a statistic for the months of November and December. It said that the probability that it would rain on a windy day is 0.3 and the probability that it would rain on a non-windy day is 0.8. The probability of a day being windy is 0.2. As a student in EECS70, you are curious to play around with these numbers. Find the probability that

i. A given day is windy and rainy.

Answer: Let R be the event that it rains on a given day and W be the event that a given day is windy. We are given $P(R|W) = 0.3$, $P(R|W^C) = 0.8$ and $P(W) = 0.2$. Then probability that a given day is both rainy and windy is $P(R \cap W) = P(R|W)P(W) = 0.3 \times 0.2 = 0.06$

ii. It rains on a given day. **Answer:** Probability that it rains on a given day is $P(R) = P(R|W)P(W) + P(R|W^C)P(W^C) = 0.3 \times 0.2 + 0.7 \times 0.8 = 0.62$

iii. Exactly one of any two days is rainy. **Answer:** Let R_1 and R_2 be the events that it rained on day 1 and day 2 respectively. Since the days are independent, $P(R_1) = P(R_2) = P(R)$. The required probability is $P(R_1)P(R_2^C) + P(R_1^C)P(R_2) = 2 * 0.62 * 0.38 = 0.4712$

iv. A non-rainy day is also non-windy. **Answer:** Probability that a non-rainy day is non-windy is $P(W^C|R^C) = \frac{P(W^C \cap R^C)}{P(R^C)} = \frac{P(R^C|W^C)(W^C)}{P(R^C)} = \frac{0.2 \times 0.8}{0.38} = \frac{8}{19}$

(h) **Chess Squares**

Two squares are chosen at random on 8×8 chessboard. What is the probability that they share a side?

Answer: In 64 squares, there are:

(1) 4 at-corner squares, each has ONLY 2 squares each having a side in common with.

(2) $6 \cdot 4 = 24$ side squares, each has ONLY 3 squares such that each has a side in common with.

(3) $6 \cdot 6 = 36$ inner squares, each has 4 squares such that each has a side in common with.

Notice that the three cases are mutually exclusive. So we just sum up the probabilities.

$\frac{4}{64} \cdot \frac{2}{63} + \frac{24}{64} \cdot \frac{3}{63} + \frac{36}{64} \cdot \frac{4}{63} = \frac{1}{18}$

5. **Short Answers**

- Each subpart is 2 point. 20 points total.
- For each problem, briefly justify your answer.

(a) For any probability space, show that $Pr[A \setminus B] \geq Pr[A] - Pr[B]$.

Answer:

$$\begin{aligned} RHS &= P[A] - P[B] && (4)\\ &= (P[A \cap B] + P[A \setminus B]) - (P[A \cap B] + P[B \setminus A]) && (5)\\ &= P[A \setminus B] - P[B \setminus A] && (6)\\ &\leq P[A \setminus B] && (7)\\ & && (8) \end{aligned}$$

(b) Show that $Pr[A \cap B] = Pr[A] + Pr[B] - Pr[A \cup B]$.

Answer:

$$\begin{aligned} RHS &= (P[A \setminus B] + P[A \cap B]) + (P[B \setminus A] + P[A \cap B]) - (P[A \setminus B] + P[B \setminus A] + P[A \cap B]) && (9)\\ &= P[A \cap B] && (10) \end{aligned}$$

(c) Assume that $|\Omega| = n$. How many distinct events does the probability space have?

Answer: There's a bijection between events and binary strings of length n. Thus, there are 2^n possible events.

(d) Assume that $|\Omega| = n$. What is the maximum number of distinct values of $Pr[A]$ can one have for events of the probability space?

Answer:

Given there are at most 2^n events, we can not achieve more than 2^n values.

We can construct 2^n values as follows:

Let the points be $p_0, \ldots, p_{n-1}$ where $P[p_i] = \frac{2^i}{2^n - 1}$.

Then, for all $0 \leq k \leq 2^n - 1$, to achieve $\frac{k}{2^n - 1}$, construct a set S_k where $p_i \in S_k$ iff the binary representation of k has a 1 at bit i.

(e) Can you find a probability space and two events A and B such that $Pr[A|B] = Pr[B]$ and A and B are not independent?

Answer:

Let $\Omega = \{1,2,3,4\}$.

Let $A = \{1\}$.

Let $B = \{1,2\}$.

$P[A|B] = 1/2 = P[B]$.

A and B are not independent: $P[A \cap B] = 1/4 \neq P[A] * P[B]$.

(f) Prove that $Pr[A \cap B \cap C] = Pr[A]Pr[B|A]Pr[C|A \cap B]$.

Answer:

$P[A \cap B \cap C] = P[C|A \cap B]P[A \cap B] = P[C|A \cap B]P[B|A]P[A]$

(g) Find an example where $Pr[A \cap B \cap C] \neq Pr[A]Pr[B|A]Pr[C|B]$.

Answer:

Let $\Omega = \{1,2\}$.

Let $B = \{1,2\}$.

Let $A = \{1\}$.

Let $C = \{2\}$.

LHS is 0. RHS is 1/4.

(h) Find an example where $Pr[A|B] > A$, another where $Pr[A|B] < Pr[A]$, and one where $Pr[A|B] = Pr[A]$.

Answer:

$Pr[A|B] > Pr[A]$: $\Omega = \{1,2\}$. $A = \{1\}$, $B = \{1\}$.

$Pr[A|B] < Pr[A]$: $\Omega = \{1,2\}$. $A = \{1\}$, $B = \{2\}$.

$Pr[A|B] = Pr[A]$: $\Omega = \{1,2\}$. $A = \{1\}$, $B = \{1,2\}$.

(i) Can you find an example where $Pr[A] > Pr[B]$ and $Pr[A|C] < Pr[B|C]$?

Answer:

```
Omega = {1, 2, 3}
A = {1, 2}
B = {3}
C = {3}
```

(j) Can you find an example where $Pr[A] > Pr[B]$ and $Pr[C|A] < Pr[C|B]$?

Answer:

```
Omega = {1, 2, 3}
A = {1, 2}
B = {3}
C = {3}
```

CS 70 Discrete Mathematics and Probability Theory
Spring 2016 Rao and Walrand HW 9

Due Thursday March 31 at 10PM

1. **Balls and Bins**You have n empty bins and you throw balls into them one by one randomly. A collision is when a ball is thrown into a bin which already has another ball.

   (a) What is the probability that the first ball thrown will cause the first collision?

   Answer: 0

   (b) What is the probability that the second ball thrown will cause the first collision? **Answer:** $\frac{1}{n}$

   (c) What is the probability that, given the first two balls are not in collision, the third ball thrown will cause the first collision? **Answer:** $\frac{2}{n}$

   (d) What is the probability that the third ball thrown will cause the first collision? **Answer:** Basically: $P(\text{Ball 3 collides} \mid \text{Ball 1, 2 do not collide}) \cdot P(\text{Ball 1, 2 do not collide})$, which is $\frac{2}{n} \cdot \frac{n-1}{n}$.

   (e) What is the probability that, given the first $m-1$ balls are not in collision, the m^{th} ball thrown will cause the first collision? **Answer:** $\frac{m-1}{n}$

   (f) What is the probability that the m^{th} ball thrown will cause the first collision? **Answer:** Similar to (d), $\frac{m-1}{n} \cdot \frac{n-1}{n} \cdot \frac{n-2}{n} \cdot \ldots \cdot \frac{n-m+2}{n} = \frac{m-1}{n} \cdot \prod_{i=0}^{m-2} \frac{n-i}{n}$.

2. **Flipping coins**

   (a) You have a fair coin, and you flip it 4 times. What is the probability that the number of heads is always ahead of the number of tails in the 4 flips? For example, the sequence HHHT has fits the description, but HTTH does not

   Answer: Let X_i denote the outcome of the i^{th} toss. X_1 must be H, since otherwise you have 1 head and 0 tails. T X_2 must also be H, since otherwise you have 1 head and 1 tail, so #heads is not strictly ahead. If X_3 is H, then X_4 can be either H or T. If X_3 is T, then X_4 must be H (otherwise #heads is not strictly ahead)
   So total probability is

   $$P(X_1 = H)P(X_2 = H)(P(X_3 = H)(P(X_4 = H) + P(X_4 = T)) + P(X_3 = T)(P(X_4) = H))$$

   which evaluates to

   $$\frac{1}{2} \cdot \frac{1}{2}(\frac{1}{2}(\frac{1}{2} + \frac{1}{2}) + \frac{1}{2} \cdot \frac{1}{2}) = \frac{3}{16}$$

   (b) What is the probability of getting 4 heads out of 4 flips, given that there are at least 2 heads?

   Answer:

   $$P(4 \text{ heads}, \text{at least 2 heads}) = P(4 \text{ heads}) = (\frac{1}{2})^4 = \frac{1}{16}$$

$$
\begin{aligned}
&P(\text{at least 2 heads}) \\
=&1 - P(\text{exactly 1 head}) - P(\text{exactly 0 heads}) \\
=&1 - 4 \cdot (\frac{1}{2})^4 - (\frac{1}{2})^4 \\
=&1 - \frac{1}{4} - \frac{1}{16} \\
=&\frac{11}{16}
\end{aligned}
$$

Using Bayes rule

$$P(4 \text{ heads} | at least 2 heads) = \frac{P(4 \text{ heads}, \text{at least 2 heads}}{P(\text{at least 2 heads})} = \frac{1/16}{11/16} = \frac{1}{11}$$

(c) Now assume that you are given two identical looking coins, but one is fair and the other is loaded, with $P(H) = 0.6$. You pick one uniformly at random, and toss it 3 times, getting 3 heads. What is the probability that you picked the loaded coin?

Answer:

$$P(\textbf{loaded}, \textbf{3 heads}) = P(\textbf{3 heads}|\textbf{loaded})P(\textbf{loaded}) = 0.6^3 \cdot 0.5 = 0.108$$

$$P(\textbf{fair}, \textbf{3 heads}) = P(\textbf{3 heads}|\textbf{fair})P(\textbf{fair}) = 0.5^3 \cdot 0.5 = 0.0625$$

Using bayes rule

$$
\begin{aligned}
&P(\textbf{loaded}|\textbf{3 heads}) \\
=&\frac{P(\textbf{loaded}, \textbf{3 heads})}{P(\textbf{3 heads})} \\
=&\frac{P(\textbf{loaded}, \textbf{3 heads})}{P(\textbf{loaded}, \textbf{3 heads}) + P(\textbf{fair}, \textbf{3 heads})} \\
=&\frac{0.6^3 \cdot 0.5}{0.6^3 \cdot 0.5 + 0.5^4} \\
=&0.633 \quad \text{(roughly)}
\end{aligned}
$$

3. **Card Game**

A game is played with six double-sided cards. One card has "1" on one side and "2" on the other. Two cards have "2" on one side and "3" on the other. And the last three cards have "3" on one side and "4" on the other. A random card is then drawn and held in a random orientation between two players, each of whom sees only one side of the card. The winner is the one seeing the smaller number. If the card that was drawn was a "2/3" card, compute the probabilities each player thinks he/she has for winning.

Answer: For the player that sees the "2" side:

$$
\begin{aligned}
\Pr[\text{Win} \mid \text{Player sees 2}] &= \frac{\Pr[\text{Win} \cap \text{Player sees 2}]}{\Pr[\text{Player sees 2}]} = \frac{\Pr[\text{Card is "2/3"} \cap \text{Player sees 2}]}{\Pr[\text{Player sees 2}]} \\
&= \frac{\Pr[\text{Card is "2/3"}] \cdot \Pr[\text{Player sees 2} \mid \text{Card is "2/3"}]}{\Pr[\text{Card has a "2"}] \cdot \Pr[\text{The side with "2" is chosen} \mid \text{Card has a "2"}]} \\
&= \frac{\frac{2}{6} \times \frac{1}{2}}{\frac{3}{6} \times \frac{1}{2}} = \frac{2}{3}
\end{aligned}
$$

For the player that sees the "3" side:

$$\Pr[\text{Win} \mid \text{Player sees 3}] = \frac{\Pr[\text{Win} \cap \text{Player sees 3}]}{\Pr[\text{Player sees 3}]} = \frac{\Pr[\text{Card is "3/4"} \cap \text{Player sees 3}]}{\Pr[\text{Player sees 3}]}$$
$$= \frac{\Pr[\text{Card is "3/4"}] \cdot \Pr[\text{Player sees 3} \mid \text{Card is "3/4"}]}{\Pr[\text{Card has a "3"}] \cdot \Pr[\text{The side with "3" is chosen} \mid \text{Card has a "3"}]}$$
$$= \frac{\frac{3}{6} \times \frac{1}{2}}{\frac{5}{6} \times \frac{1}{2}} = \frac{3}{5}$$

4. **Boys and Girls**

There are three children in a family. A friend is told that at least two of them are boys. What is the probability that all three are boys? The friend is then told that the two are the oldest two children. Now what is the probability that all three are boys? Use Bayes' Law to explain this. Assume throughout that each child is independently either a boy or a girl with equal probability.

Answer: Let A be the information that you are told and B the event that all three are boys. By Bayes' rule, we have $\Pr[B \mid A] = \frac{\Pr[B] \cdot \Pr[A|B]}{\Pr[A]}$. However, note that whether $A =$ "at least two boys" or $A =$ "oldest two are boys", $\Pr[A \mid B]$ is simply 1. So in either case we have $\Pr[B \mid A] = \frac{\Pr[B]}{\Pr[A]}$.

Hence,

$$\Pr[\text{All 3 boys} \mid \text{At least 2 boys}] = \frac{\Pr[\text{All 3 boys}]}{\Pr[\text{At least 2 boys}]} = \frac{\frac{1}{8}}{\frac{4}{8}} = \frac{1}{4}$$

$$\Pr[\text{All 3 boys} \mid \text{Oldest 2 are boys}] = \frac{\Pr[\text{All 3 boys}]}{\Pr[\text{Oldest 2 are boys}]} = \frac{\frac{1}{8}}{\frac{2}{8}} = \frac{1}{2}.$$

In this case we saw Bayes' rule simplify to $\Pr[B \mid A] = \frac{\Pr[B]}{\Pr[A]}$. Since B is a subset of A, the formula directly shows that this conditional probability depends only on the number of possibilities contained in A. When we are told that at least two children are boys, any of the three children could be a girl. In contrast, if we are told that the oldest two children are boys, then only the youngest child has the possibility of being a girl. Therefore the latter case has fewer possibilities and therefore larger conditional probability.

5. **Birthdays**

Suppose you record the birthdays of a large group of people, one at a time until you have found a match, i.e., a birthday that has already been recorded. (Assume there are 365 days in a year.)

(a) What is the probability that it takes more than 20 people for this to occur?

Answer: $\Pr[\text{it takes more than 20 people}] = \Pr[\text{20 people don't have the same birthday}] = \frac{\frac{365!}{(365-20)!}}{365^{20}} = \boxed{\frac{\frac{365!}{345!}}{365^{20}}} \approx .589$

Another explanation that does not use counting:

Let b_i be the birthday of the i-th person.

$$\begin{aligned}
&\Pr[\text{it takes more than 20 people}] \\
=&\Pr[b_{20} \neq b_i \mid b_i\text{'s are all different } \forall 1 \leq i \leq 19] \times \Pr[b_i\text{'s are all different } \forall 1 \leq i \leq 19] \\
=&\Pr[b_{20} \neq b_i \mid b_i\text{'s are all different } \forall 1 \leq i \leq 19] \times \Pr[b_{19} \neq b_i \mid b_i\text{'s are all different } \forall 1 \leq i \leq 18] \times \\
&\cdots \times \Pr[b_3 \neq b_i \mid b_i\text{'s are all different } \forall 1 \leq i \leq 2] \times \Pr[b_2 \neq b_1] \\
=&\frac{365-19}{365} \times \frac{365-18}{365} \times \cdots \times \frac{363}{365} \times \frac{364}{365} \\
\approx&.589
\end{aligned}$$

(b) What is the probability that it takes exactly 20 people for this to occur?
Answer: $\Pr[\text{it takes exactly 20 people}] =$

$\Pr[\text{first 19 have different birthdays } \textbf{and } 20^{\text{th}}\text{person shares a birthday with one of the first 19}]$.
How total ways can the birthdays be chosen for 20 people? 365^{20}. How many ways can the birthdays be chosen so the first 19 have different birthdays and the 20^{th} person shares a birthday with the first 19? Well, the firs person has 365 choices, the second has 364 choices left, and so on until the nineteenth person has $(365-19+1) = 347$ choices left. Then, the 20^{th} person has 19 choices for his birthday. So in total, there are $365 \cdot 364 \cdot \cdots \cdot 348 \cdot 347 \cdot 19 = \frac{365!}{346!} \cdot 19$ ways of getting what we want. So $\Pr[\text{it takes exactly 20 people}] = \frac{365 \cdot 364 \cdot \cdots \cdot 348 \cdot 347 \cdot 19}{365^{20}} = \boxed{\frac{\frac{365!}{346!} \cdot \mathbf{19}}{\mathbf{365^{20}}}} \approx .032$

Another explanation that does not use counting:
Let b_i be the birthday of the i-th person.

$$\begin{aligned}
&\Pr[\text{it takes exactly 20 people}] \\
=&\Pr[b_{20} \text{ is equal to one of the } b_i\text{'s} \mid b_i\text{'s are all different } \forall 1 \leq i \leq 19] \times \\
&\Pr[b_i\text{'s are all different } \forall 1 \leq i \leq 19] \\
=&\Pr[b_{20} \text{ is equal to one of the } b_i\text{'s} \mid b_i\text{'s are all different } \forall 1 \leq i \leq 19] \times \\
&\Pr[b_{19} \neq b_i \mid b_i\text{'s are all different } \forall 1 \leq i \leq 18] \times \cdots \times \\
&\Pr[b_3 \neq b_i \mid b_i\text{'s are all different } \forall 1 \leq i \leq 2] \times \Pr[b_2 \neq b_1] \\
=&\frac{19}{365} \times \frac{365-18}{365} \times \cdots \times \frac{363}{365} \times \frac{364}{365} \\
\approx&.032
\end{aligned}$$

(c) Suppose instead that you record the birthdays of a large group of people, one at a time, until you have found a person whose birthday matches your own birthday. What is the probability that it takes exactly 20 people for this to occur?
Answer: $\Pr[\text{it takes exactly 20 people}] =$

$\Pr[\text{first 19 don't have your birthday } \textbf{and } 20^{\text{th}}\text{person has your birthday}]$.
Similar to the last problem, there are 364 choices for the first person's birthday to be different than yours, 364 for the second person, and so on until the nineteenth person has 364 choices. Then, the 20^{th} person has exactly 1 choice to have your birthday. So the total number of ways to get what we want is $364^{19} \cdot 1$. There are 365^{20} possibilities total. So $\Pr[\text{it takes exactly 20 people}] =$

$\boxed{\frac{364^{19}}{365^{20}}} \approx .0026$

Another explanation that does not use counting:

$$\begin{aligned}
\Pr[\text{it takes exactly 20 people}] =& \Pr[\text{the 1st person does not have the same birthday as yours}] \times \\
& \Pr[\text{the 2nd person does not have the same birthday as yours}] \times \\
& \cdots \times \Pr[\text{the 19th person does not have the same birthday as yours}] \times \\
& \Pr[\text{the 20th person has the same birthday as yours}] \\
=& \frac{364}{365} \times \frac{364}{365} \times \cdots \frac{364}{365} \times \frac{1}{365} \\
=& \frac{364^{19} \times 1}{365^{20}} \\
\approx& 0.0026
\end{aligned}$$

6. **(Alvin's woes)** After a long night of debugging, Alvin has just perfected the new homework party/office hour queue system. CS 70 students sign themselves up for the queue, and TAs go through the queue, resolving requests one by one. Unfortunately, our newest TA (let's call him TA Bob) does not understand how to use the new queue: instead of resolving the requests in order, he always uses the Random Student button, which (as the name suggests) chooses a random student in the queue for him. To make matters worse, after helping the student, Bob forgets to click the Resolve button, so the student still remains in the queue! For this problem, assume that there are n total students in the queue.

   (a) Suppose that Bob has already helped k students. What is the probability that the Random Student button will take him to a student who has not already been helped?

   (b) Give a description of the probability space Ω. Fully answering this question entails giving a representation for an outcome ω, as well as any conditions on what ω is allowed to be. As an example, the probability space for two coin flips can be described as $\{HH, HT, TH, TT\}$. Each outcome ω is a length-two string of characters from the set $\{H, T\}$, where H represents heads and T represents tails; the first character of ω is the result of the first coin flip, and the second character is the result of the second coin flip. (Hint: Each outcome ω should include information about which students have been helped, along with the total number of Random Student button presses.)

   (c) Let X_i^r be the event that TA Bob has not helped student i after pressing the Random Student button a total of r times. What is $\Pr[X_i^r]$? Assume that the results of the Random Student button are independent of each other. Now approximate the answer using the inequality $1 - x \leq e^{-x}$.

   (d) Let T_r represent the event that TA Bob presses the Random Student button r times, but still has not been able to help all n students. (In other words, it takes TA Bob longer than r Random Student button presses before he manages to help every student). What is T_r in terms of the events X_i^r? (Hint: Events are subsets of the probability space Ω, so you should be thinking of set operations...)

   (e) Using your answer for the previous part, what is an upper bound for $\Pr[T_r]$? (You may leave your answer in terms of $\Pr[X_i^r]$. Use the inequality $1 - x \leq e^{-x}$ from before.)

   (f) Now let $r = \alpha n \ln n$. What is $\Pr[X_i^r]$?

(g) Calculate an upper bound for $\Pr[T_r]$ using the same value of r as before. (This is more formally known as a bound on the tail probability of the distribution of button presses required to help every student. This distribution will be explored in more detail later, in the context of random variables.)

(h) What value of r do you need to bound the tail probability by $1/n^2$? In other words, how many button presses are needed so that the probability that TA Bob has not helped every student is at most $1/n^2$?

Answer: Solution

(a) There are $n-k$ students who have not been helped, and the probability that one of these students is chosen is $(n-k)/n$ or $1-k/n$.

(b) One way to describe Ω is the set of all tuples of length $n+1$, where the first n positions are either 0 or 1 (representing whether the student at position k in the queue has been helped or not) and the last position is an element of $\mathbb{N}$ (representing the total number of button presses). Additionally, ω must satisfy the condition that the number of students helped is less than or equal to the total number of button presses. This question was intended to solidify your understanding of probability spaces and their representations for complicated problems.

(c) The probability that student i is chosen by the Random Student button is $1/n$, so the complement of this probability is $1-1/n$. Using the assumption of independence:

$$\Pr[X_i^r] = \left(1-\frac{1}{n}\right)^r \leq e^{-r/n}$$

(d) T_r is the event that TA Bob has pressed the button r times, but has not been able to help either student 1, or student 2, or student 3, ... This is the union: $T_r = \bigcup_{i=1}^{n} X_i^r$.

(e) Use the union bound. $\Pr[T_r] \leq n \cdot \Pr[X_i^r] \leq ne^{-r/n}$.

(f) Plug in for r. $\Pr[X_i^r] \leq e^{-r/n} = e^{-\alpha \ln n} = n^{-\alpha}$.

(g) A quick application of the union bound derived in the previous parts: $\Pr[T_r] \leq n \cdot \Pr[X_i^r] = n^{1-\alpha}$.

(h) Set $1-\alpha = -2$, which is $\alpha = 3$. This gives $r = 3n\ln n$. (Side-note: This problem is more commonly known as the coupon collector's problem. Once we cover random variables, we will see that the expected number of button presses required to help every student is $\Theta(n \log n)$.)