CS 70 Discrete Mathematics and Probability Theory
Summer 2023 Huang, Suzani, and Tausik

HW 1

1 Proof by?

Note 2

(a) Prove that for any two integers x and y, if 10 does not divide xy, then 10 does not divide x and 10 does not divide y. In notation: $(\forall x, y \in \mathbb{Z})\ (10 \nmid xy) \implies ((10 \nmid x)\ \wedge\ (10 \nmid y))$. What proof technique did you use?

(b) Prove or disprove the contrapositive.

(c) Prove or disprove the converse.

Solution:

(a) We will use proof by contraposition. For any arbitrary given x and y, the statement $(10 \nmid xy) \implies ((10 \nmid x) \wedge (10 \nmid y))$ is equivalent using contraposition to $\neg(10 \nmid x \wedge 10 \nmid y) \implies \neg(10 \nmid xy)$. Moving the negations inside, this becomes equivalent to $(10 \mid x \vee 10 \mid y) \implies 10 \mid xy$.

Now for this part, we give a proof by cases. Assuming that $10 \mid x \vee 10 \mid y$, one of the two cases must be true.

(a) $10 \mid x$: in this case $x = 10k$ for some $k \in \mathbb{Z}$. Therefore $xy = 10ky$ which is a multiple of 10. So $10 \mid xy$.

(b) $10 \mid y$: in this case $y = 10k$ for some $k \in \mathbb{Z}$. Therefore $xy = 10kx$ which is a multiple of 10. So $10 \mid xy$.

Therefore assuming $10 \mid x \vee 10 \mid y$ we proved $10 \mid xy$.

We used proof by cases and proof by contraposition.

(b) We proved the statement. The contrapositive of a statement is logically equivalent to the statement. So we are done.

(c) It's not true! The converse is that if 10 does not divide x and does not divide y then 10 does not divide xy. We can choose $x = 2$ and $y = 5$ and see a counterexample to the statement.

2 Prove or Disprove

Note 2

For each of the following, either prove the statement, or disprove by finding a counterexample.

(a) $(\forall n \in \mathbb{N})$ if n is odd then $n^2 + 4n$ is odd.

(b) $(\forall a, b \in \mathbb{R})$ if $a + b \leq 15$ then $a \leq 11$ or $b \leq 4$.

(c) $(\forall r \in \mathbb{R})$ if r^2 is irrational, then r is irrational.

(d) $(\forall n \in \mathbb{Z}^+)$ $5n^3 > n!$. (Note: $\mathbb{Z}^+$ is the set of positive integers)

Solution:

(a) **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 $n^2 + 4n$, we get $(2k + 1)^2 + 4 \cdot (2k + 1)$. Simplifying the expression yields $4k^2 + 12k + 5$. This can be rewritten as $2 \cdot (2k^2 + 6k + 2) + 1$. Since $2k^2 + 6k + 2$ is a natural number, by the definition of odd numbers, $n^2 + 4n$ is odd.
Alternatively, we could also factor the expression to get $n(n + 4)$. Since n is odd, $n + 4$ is also odd. The product of 2 odd numbers is also an odd number. Hence $n^2 + 4n$ is odd.

(b) **Answer**: True.
Proof: We will use a proof by contraposition. Suppose that $a > 11$ and $b > 4$ (note that this is equivalent to $\neg(a \leq 11 \vee b \leq 4)$). Since $a > 11$ and $b > 4$, $a + b > 15$ (note that $a + b > 15$ is equivalent to $\neg(a + b \leq 15)$). Thus, if $a + b \leq 15$, then $a \leq 11$ or $b \leq 4$.

(c) **Answer**: True.
Proof: We will use a proof by contraposition. Assume that r is rational. Since r is rational, it can be written in the form $\frac{a}{b}$ where a and b are integers with $b \neq 0$. Then r^2 can be written as $\frac{a^2}{b^2}$. By the definition of rational numbers, r^2 is a rational number, since both a^2 and b^2 are integers, with $b \neq 0$. By contraposition, if r^2 is irrational, then r is irrational.

(d) **Answer**: False.
Proof: We will show a counterexample. Let $n = 7$. $5 \cdot 3^3 = 1715$. $7! = 5040$. Since $5n^3 < n!$, the claim is false.

3 Proving Inequality

Note 3

For all positive integers $n \geq 1$, prove that

$$\frac{1}{3^1} + \frac{1}{3^2} + \ldots + \frac{1}{3^n} < \frac{1}{2}.$$

(Note: while you can use formula for an infinite geometric series to prove this, we would like you to use induction. If you're having trouble with the inductive step, try strengthening the inductive hypothesis. Can you prove an equality statement instead of an inequality?)

Solution:

Show that induction based on this claim doesn't get us anywhere. Try a few cases and come up with a stronger inductive hypothesis. For example:

- $\frac{1}{3} = \frac{1}{2} - \frac{1}{6}$
- $\frac{1}{3} + \frac{1}{9} = \frac{1}{2} - \frac{1}{18}$
- $\frac{1}{3} + \frac{1}{9} + \frac{1}{27} = \frac{1}{2} - \frac{1}{54}$

One possible statement is

$$\frac{1}{3^1} + \frac{1}{3^2} + \ldots + \frac{1}{3^n} = \frac{1}{2} - \frac{1}{2 \cdot 3^n}$$

- *Base Case:* $n = 1$. $\frac{1}{3} = \frac{1}{2} - \frac{1}{6}$. True.
- *Inductive Hypothesis:* Assume the statement holds for $n \geq 1$.
- *Inductive Step:* Starting from the left hand side,

$$\begin{aligned} \frac{1}{3^1} + \frac{1}{3^2} + \ldots + \frac{1}{3^n} + \frac{1}{3^{n+1}} &= \frac{1}{2} - \frac{1}{2 \cdot 3^n} + \frac{1}{3^{n+1}} \\ &= \frac{1}{2} - \frac{3-2}{2 \cdot 3^{n+1}} \\ &= \frac{1}{2} - \frac{1}{2 \cdot 3^{n+1}}. \end{aligned}$$

Therefore, $\frac{1}{3^1} + \frac{1}{3^2} + \ldots + \frac{1}{3^n} = \frac{1}{2} - \frac{1}{2 \cdot 3^n} < \frac{1}{2}$.

4 A Coin Game

Note 3

Your "friend" Stanley Ford suggests you play the following game with him. You each start with a single stack of n coins. On each of your turns, you select one of your stacks of coins (that has at least two coins) and split it into two stacks, each with at least one coin. Your score for that turn is the product of the sizes of the two resulting stacks (for example, if you split a stack of 5 coins into a stack of 3 coins and a stack of 2 coins, your score would be $3 \cdot 2 = 6$). You continue taking turns until all your stacks have only one coin in them. Stan then plays the same game with his stack of n coins, and whoever ends up with the largest total score over all their turns wins.

Prove that no matter how you choose to split the stacks, your total score will always be $\frac{n(n-1)}{2}$. (This means that you and Stan will end up with the same score no matter what happens, so the game is rather pointless.)

Solution:

We can prove this by strong induction on n.

Base Case: If $n = 1$, you start with a stack of one coin, so the game immediately terminates. Your total score is zero–and indeed, $\frac{n(n-1)}{2} = \frac{1 \cdot 0}{2} = 0$.

Inductive Step: Suppose that if you start with i coins (for i between 1 and n inclusive), your score will be $\frac{i(i-1)}{2}$ no matter what strategy you employ. Now suppose you start with $n+1$ coins. In your first move, you must split your stack into two smaller stacks. Call the sizes of these stacks s_1 and s_2 (so $s_1 + s_2 = n+1$ and $s_1, s_2 \geq 1$). Your end score comes from three sources: the points you get from making this first split, the points you get from future splits involving coins from stack 1, and the points you get from future splits involving coins from stack 2. From the rules of the game, we know you get $s_1 s_2$ points from the first split. From the inductive hypothesis (which we can apply because s_1 and s_2 are between 1 and n), we know that the total number of points you get from future splits of stack 1 is $\frac{s_1(s_1-1)}{2}$ and similarly that the total number of points you get from future splits of stack 2 is $\frac{s_2(s_2-1)}{2}$, regardless of what strategy you employ in splitting them. Thus, the total number of points we score is

$$\begin{aligned} s_1 s_2 + \frac{s_1(s_1-1)}{2} + \frac{s_2(s_2-1)}{2} &= \frac{s_1(s_1-1) + 2s_1 s_2 + s_2(s_2-1)}{2} \\ &= \frac{\big(s_1(s_1-1) + s_1 s_2\big) + \big(s_2(s_2-1) + s_1 s_2\big)}{2} \\ &= \frac{s_1(s_1+s_2-1) + s_2(s_1+s_2-1)}{2} \\ &= \frac{(s_1+s_2)(s_1+s_2-1)}{2} \end{aligned}$$

Since $s_1 + s_2 = n+1$, this works out to $\frac{(n+1)(n+1-1)}{2}$, which is what we wanted to show your total number of points came out to. This completes our proof by induction.

5 Nothing Can Be Better Than Something

Note 4

In the stable matching problem, suppose that some jobs and candidates have hard requirements and might not be able to just settle for anything. In other words, each job/candidate prefers being unmatched rather than be matched with those below a certain point in their preference list. Let the term "entity" refer to a candidate/job. A matching could ultimately have to be partial, i.e., some entities would and should remain unmatched.

Consequently, the notion of stability here should be adjusted a little bit to capture the autonomy of both jobs to unilaterally fire employees and/or employees to just walk away. A matching is stable if

- there is no matched entity who prefers being unmatched over being with their current partner;
- there is no matched/filled job and unmatched candidate that would both prefer to be matched with each other over their current status;
- there is no matched job and matched candidate that would both prefer to be matched with each other over their current partners; and
- similarly, there is no unmatched job and matched candidate that would both prefer to be matched with each other over their current status;
- there is no unmatched job and unmatched candidate that would both prefer to be with each other over being unmatched.

(a) Prove that a stable pairing still exists in the case where we allow unmatched entities.

(HINT: You can approach this by introducing imaginary/virtual entities that jobs/candidates "match" if they are unmatched. How should you adjust the preference lists of jobs/candidates, including those of the newly introduced imaginary ones for this to work?)

(b) As you saw in the lecture, we may have different stable matchings. But interestingly, if an entity remains unmatched in one stable matching, they must remain unmatched in any other stable matching as well. Prove this fact by contradiction.

Solution:

(a) We form an instance of the standard stable matching problem as follows. Following, the hint, we introduce an imaginary mate, r_e, (let's call it a robot) for each entity. Note that we introduce one robot for each entity, i.e. there are as many robots as there are candidates+jobs. For simplicity let us say each robot, r_e, is owned by the entity, e, we introduce it for.

Each robot, r_e, prefers its owner e, i.e. it puts its owner at the top of its preference list. The rest of its preference list is arbitrary and includes all other owners and the robots of other types of entities. An entity e of a robot, r_e puts it in their preference list exactly after the last entity they are willing to match with. i.e. owners like their robots more than entities they are not willing to match, but less than entities they like to match. The other robots can be placed in arbitrary order in e's list.

For this instance of the stable matching problem which we refer to as the robot instance, I, the propose and reject algorithm will give us a stable matching, M.

To extract the desired partial matching, we simply remove all pairs in M with at least one robot (two robots can match each other). We refer to each entity which is not matched as single. We observe, an entity, e, will never be matched with another entity's robot, $r_{e'}$, because then e and its robot, r_e, would form a rogue couple (r_e prefers e to other owners, and e prefers r_e more than other robots). It remains to show this is a stable matching as specified in the problem as follows:

- there is no matched entity, e, who prefers being unmatched over being with their current partner since the e and r_e would form a rogue couple in M;
- there is no matched/filled job, j, and unmatched candidate, c, that would both prefer to be matched with each other since otherwise j and c would form a rogue couple in M;
- there is no matched job j and matched candidate c that would both prefer to be matched with each other over their current partners since then j and c would form a rogue couple in M;
- similarly, there is no unmatched job, j, and matched candidate, c, that would both prefer to be matched with each other over their current status since j and c would be a rogue couple in M;
- there is no unmatched job, j, and unmatched candidate, c, that would both prefer to be with each other over being unmatched since j and c would be a rogue couple in M.

Thus, the resulting partial matching is stable.

(b) We will perform proof by contradiction. Assume that there exists some job j_1 who is paired with a candidate c_1 in stable pairing S and unpaired in stable pairing T. The stable pairing S where j_1 and c_1 are paired means j_1 and c_1 both prefer to be with each other over being single. Since T is a stable pairing and j_1 is unpaired, c_1 must be paired in T with a job j_2 whom they prefer over j_1. (If c_1 were unpaired or paired with a job they do not prefer over j_1, then (j_1, c_1) would be a rogue couple in T, which is a contradiction.)

Since j_2 is paired with c_1 in T, it must be paired in S with some candidate c_2 whom j_2 prefers over c_1. This process continues (c_2 must be paired with some j_3 in T, j_3 must be paired with some c_3 in S, etc.) with the pattern that (c_i, j_i) is in S and (j_i, c_{i-1}) is in T. At some time i, one must encounter j_1 in this process as there are a finite number of jobs. A this point $(j_1 = j_i, c_{i-1})$ is in T, which implies that j_1 is matched in T.

A similar argument can be used for candidates.

This contradicts the assumption that j_1 is unmatched in T. Since no job or candidate can be paired in one stable pairing and unpaired in another, every job or candidate must be either paired in all stable pairings or unpaired in all stable pairings.

6 Universal Preference

Note 4

Suppose that preferences in a stable matching instance are universal: all n jobs share the preferences $C_1 > C_2 > \cdots > C_n$ and all candidates share the preferences $J_1 > J_2 > \cdots > J_n$.

(a) What pairing do we get from running the algorithm with jobs proposing? Can you prove this happens for all n?

(b) What pairing do we get from running the algorithm with candidates proposing?

(c) What does this tell us about the number of stable pairings?

Solution:

(a) The pairing results in (C_i, J_i) for each $i \in \{1, 2, ..., n\}$. This result can be proved by induction:

Our base case is when $n = 1$, so the only pairing is (C_1, J_1), and thus the base case is trivially true.

Now assume this is true for some $n \in \mathbb{N}$. On the first day with $n+1$ jobs and $n+1$ candidates, all $n+1$ jobs will propose to C_1. C_1 prefers J_1 the most, and the rest of the jobs will be rejected. This leaves a set of n unpaired jobs and n unpaired candidates who all have the same preferences (after the pairing of (C_1, J_1)). By the process of induction, this means that every i^{th} preferred candidate will be paired with the i^{th} preferred job.

(b) The pairings will again result in (J_i, C_i) for each $i \in \{1, 2, ..., n\}$. This can be proved by induction in the same as above, but replacing "job" with "candidate" and vice-versa.

(c) We know that job-proposing produces a candidate-pessimal stable pairing. We also know that candidate-proposing produces a candidate-optimal stable pairing. We found that candidate-optimal and candidate-pessimal pairings are the same. This means that there is only one stable pairing, since both the best and worst pairings (for candidates) are the same pairings.

7 Preserving Set Operations

Note 0
Note 2

For a function f, define the image of a set X to be the set $f(X) = \{y \mid y = f(x) \text{ for some } x \in X\}$. Define the inverse image or preimage of a set Y to be the set $f^{-1}(Y) = \{x \mid f(x) \in Y\}$. Prove the following statements, in which A and B are sets.

Recall: For sets X and Y, $X = Y$ if and only if $X \subseteq Y$ and $Y \subseteq X$. To prove that $X \subseteq Y$, it is sufficient to show that $(\forall x)\ ((x \in X) \implies (x \in Y))$.

(a) $f^{-1}(A \cup B) = f^{-1}(A) \cup f^{-1}(B)$.

(b) $f(A \cup B) = f(A) \cup f(B)$.

Solution:

In order to prove equality $A = B$, we need to prove that A is a subset of B, $A \subseteq B$ and that B is a subset of A, $B \subseteq A$. To prove that LHS is a subset of RHS we need to prove that if an element is a member of LHS then it is also an element of the RHS.

(a) Suppose $x \in f^{-1}(A \cup B)$ which means that $f(x) \in A \cup B$. Then either $f(x) \in A$, in which case $x \in f^{-1}(A)$, or $f(x) \in B$, in which case $x \in f^{-1}(B)$, so in either case we have $x \in f^{-1}(A) \cup f^{-1}(B)$. This proves that $f^{-1}(A \cup B) \subseteq f^{-1}(A) \cup f^{-1}(B)$.

Now, suppose that $x \in f^{-1}(A) \cup f^{-1}(B)$. Suppose, without loss of generality, that $x \in f^{-1}(A)$. Then $f(x) \in A$, so $f(x) \in A \cup B$, so $x \in f^{-1}(A \cup B)$. The argument for $x \in f^{-1}(B)$ is the same. Hence, $f^{-1}(A) \cup f^{-1}(B) \subseteq f^{-1}(A \cup B)$.

(b) Suppose that $x \in A \cup B$. Then either $x \in A$, in which case $f(x) \in f(A)$, or $x \in B$, in which case $f(x) \in f(B)$. In either case, $f(x) \in f(A) \cup f(B)$, so $f(A \cup B) \subseteq f(A) \cup f(B)$.

Now, suppose that $y \in f(A) \cup f(B)$. Then either $y \in f(A)$ or $y \in f(B)$. In the first case, there is an element $x \in A$ with $f(x) = y$; in the second case, there is an element $x \in B$ with $f(x) = y$. In either case, there is an element $x \in A \cup B$ with $f(x) = y$, which means that $y \in f(A \cup B)$. So $f(A) \cup f(B) \subseteq f(A \cup B)$.

A common pitfall for this question is to start with an element $y \in f(A \cup B)$, and to take $f^{-1}(y) \in A \cup B$. The issue here is that $f^{-1}(y)$ is not necessarily a single element; it can be a set of elements, so the more precise statement is $f^{-1}(\{y\}) \subseteq A \cup B$. Here, we can't necessarily conclude that either $f^{-1}(\{y\}) \subseteq A$ or $f^{-1}(\{y\}) \subseteq B$, since $f^{-1}(\{y\})$ could contain some elements in A and some elements in B. This would require more careful consideration; it's easier in this case to work with an element $x \in A \cup B$.

The purpose of this problem is to gain familiarity to naming things precisely. In particular, we named an element in the LHS (or the pre-image of the LHS) and then argued about whether that element or its image was in the right hand side. By explicitly naming an element generically where it could be *any* element in the set, we could argue about its membership in a set and or its image or preimage. With these different concepts floating around it is helpful to be clear in the argument.