CS 70 Discrete Mathematics and Probability Theory
Fall 2016 Seshia and Walrand **Discussion 2A**

Note: There will be no sections covering this worksheet. But we highly recommend finishing the discussion worksheet on your own time. It will help you with the upcoming homework.Ask questions in office hours if you have any.

1. **Proving Inequality**
For all positive integers $n \geq 1$, prove that

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

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

2. **True/False**

If $n > 0$ is a positive integer, and S is a set of distinct positive integers, all of which are less than or equal to n, then S has at most n elements.

Solution: The implication is $\boxed{\text{TRUE}}$

(a) **Proof by Induction**

Base Case: For $n = 1$, S cannot contain any element > 1. Therefore S can only be: $\emptyset$, or $\{1\}$. In both cases, is true that $|S| \leq n = 1$.

Inductive Hypothesis: Assume that for $n = k$, **all sets** S for which $\forall x \in S : 1 \leq x \leq k$, satisfy $|S| \leq k$.

Induction Step: For $n = k+1$, consider a set S such that $\forall x \in S : 1 \leq x \leq k+1$. Consider partitioning S into two sets A and B, such that $A = \{x \in S : 1 \leq x \leq k\}$ and $B = \{x \in S : x > k\}$. By the hypothesis, $|A| \leq k$. Further, by definition we have $\forall x \in B : k < x \leq k+1$. Thus B cannot contain any element $x \in \mathbb{Z}, x \neq k+1$. So B can only be $\emptyset$ or $\{k+1\}$. In both cases, $|B| \leq 1$. Since A and B partition S, we have $|S| = |A| + |B| \leq k+1$.

(b) **Proof by Contradiction**

Suppose there exists a set S such that $\forall x \in S : 1 \leq x \leq n$ but $|S| > n$. Apply the Pigeonhole Principle: There are n positive integers between 1 and n (boxes), and each distinct $x \in S$ (pigeons) must be placed in one of these boxes. By the Pigeonhole Principle, there must exist some two pigeons in the same box – or some two elements x_1, x_2 that are not distinct. Contradiction.

(c) **Direct Proof**

Let $A_n = \{x \in \mathbb{Z} : 1 \leq x \leq n\}$. By explicit enumeration, A_n contains exactly n distinct elements, so $|A_n| = n$. Then any set S such that $\forall x \in S : 1 \leq x \leq n$ is a subset of A_n. And $S \subseteq A_n \implies |S| \leq |A_n| = n$.

All of the above proofs were accepted, though the first is most rigorous. In fact, the proving the Pigeonhole Principle, and proving (rigorously) that $A \subseteq B \implies |A| \leq |B|$ requires a very similar inductive argument.

This problem statement is so intuitively obvious that many people accidentally assumed the statement in their proof. For example, in the inductive step we cannot say something like "remove the maximum element of S", since technically we do not know that S is finite (without assuming the statement itself). We did not require such levels of rigor in grading this problem, but such things are good to keep in mind.

3. **More squares**

Prove the following statement: $\forall n \in \mathbb{N}, \exists m \in \mathbb{N}$ such that $\sum_{k=0}^{n} k^3 = m^2$

Solution: Let's try normal induction:
Assume $\sum_{k=0}^{n} k^3 = m^2$ for some m.
Consider $n+1$ case: $\sum_{k=0}^{n+1} k^3 = m^2 + (n+1)^3$. (We will get stuck here!)

The easier problem would have been to give m directly, namely, ask to prove

$$\sum_{k=0}^{n} k^3 = (\sum_{k=0}^{n} k)^2$$

The idea behind not giving m out is to have them think of strengthening the induction hypothesis, by constructing m itself, which in turn forces them to try the expression out with a few n.

The heart of the reasoning should read something like:

$$\begin{aligned}
\sum_{k=0}^{n+1} k^3 &= (n+1)^3 + (\sum_{k=0}^{n} k^3) \\
&= (n+1)^3 + (\sum_{k=0}^{n} k)^2 \\
&= \left((n+1) + (\sum_{k=0}^{n} k)\right)^2 - \left((n+1)^2 + 2(n+1)(\sum_{k=0}^{n} k)\right) + (n+1)^3 \\
&= (\sum_{k=0}^{n+1} k)^2 - \left((n+1)^2 + 2(n+1)\frac{n(n+1)}{2}\right) + (n+1)^3 \\
&= (\sum_{k=0}^{n+1} k)^2 - (n+1)^3 + (n+1)^3 \\
&= (\sum_{k=0}^{n+1} k)^2
\end{aligned}$$

4. **Stable Marriage**

Consider the set of men $M = \{1, 2, 3\}$ and the set of women $W = \{A, B, C\}$ with the following preferences.

Men	Women		
1	A	B	C
2	B	A	C
3	A	B	C

Women	Men		
A	2	1	3
B	1	2	3
C	1	2	3

Run the male propose-and-reject algorithm on this example. How many days does it take and what is the resulting pairing? (Show your work)

Solution: The algorithm takes 3 days to produce a matching. The resulting pairing is {(A, 1), (B, 2), (C, 3)}

Woman	Day 1	Day 2	Day 3
A	(1),3	(1)	(1)
B	(2)	(2),3	(2)
C			(3)

5. **Propose-and-Reject Proofs**

Prove the following statements about the traditional propose-and-reject algorithm.

(a) In any execution of the algorithm, if a woman receives a proposal on day i, then she receives some proposal on every day thereafter until termination.

Solution: The idea is to use the Improvement Lemma (remind students of the Improvement Lemma). If a woman receives a proposal on day i, then, by the Improvement Lemma, she will always have someone as good or better proposing to her every day after day i.

(b) In any execution of the algorithm, if a woman receives no proposal on day i, then she receives no proposal on any previous day j, $1 \leq j < i$.

Solution: One way is to use a proof by contradiction. Assume that a woman receives no proposal on day i but did receive a proposal on some previous day j, $1 \leq j < i$. By the Improvement Lemma, since the woman received a proposal on day j, then she will always have someone as good or better proposing to her every day after day j. But then the woman must receive a proposal on day $i > j$. Contradiction.

(c) In any execution of the algorithm, there is at least one woman who only receives a single proposal (Hint: use the parts above!)

Solution: Let's say the algorithm takes k days - then we know that every woman receives a proposal on day k. There is at least one woman w who does not receive a proposal on day $k-1$. (You will prove this in your homework!) Then from part (b), since w did not receive a proposal on day $k-1$, she didn't receive a proposal on any day before k. Since w was not proposed to on days $1,\dots,k-1$ and is proposed to on day k (since we know this is when the algorithm terminates) then w receives only one proposal.