Mod 1: Cardinality

Rushil Umaretiya

frj2ka@virginia.edu

#1 For each of the following claims, state whether it is true or false and then prove your assertion.

1. All finite sets have an injection to $\mathbb{N}$

   True. Let A be a finite set. Then, $|A| = n$ for some $n \in \mathbb{N}$. Let $f : A \to \mathbb{N}$ be defined as $f(a) = n$ for all $a \in A$. This function is injective because for all $a, b \in A$, if $f(a) = f(b)$, then $n = n$, which implies that $a = b$. Thus, all finite sets have an injection to $\mathbb{N}$ when we map each element of A to a unique natural number.

2. All finite sets have a surjection to $\mathbb{N}$

   False. Let A be a finite set with $|A| = n$. Then, the maximum number of elements in the image of a function $f : A \to \mathbb{N}$ is n. Suppose there exists a surjection $f : A \to \mathbb{N}$. Then, the image of f must contain all natural numbers, which is not possible because the image of f is finite. Thus, all finite sets do not have a surjection to $\mathbb{N}$.

3. If A is a countably infinite set (i.e., $|A| = |\mathbb{N}|$) and B is a also a countably infinite set (i.e., $|B| = |\mathbb{N}|$), then $A \cup B$ is also countable.

   True. If we assign the elements of A to the even values of $\mathbb{N}$ and the elements of B to the odd values of $\mathbb{N}$, then we can create a bijection between $A \cup B$ and $\mathbb{N}$ as each element of $A \cup B$ is mapped to a unique element in $\mathbb{N}$. Thus, $A \cup B$ is countable.

4. If A is countably infinite and B is uncountably infinite, then $A \cup B$ is countable.

   False. Note that $B \subseteq A \cup B$. Since B is uncountably infinite, then $A \cup B$ must also be uncountably infinite because the union of a countably infinite set and an uncountably infinite set is uncountably infinite. Thus, $A \cup B$ is uncountable.

5. If A is countably infinite and B is uncountably infinite, then $A \cap B$ is countable.

   True. Since A is countably infinite, we can list the elements of A as $a_1, a_2, a_3, \ldots$. Since B is uncountably infinite, there must be at least one element in B that is not in A. Thus, $A \cap B$ is finite or countably infinite. Since $A \cap B$ is a subset of A, it must be countable.

#2 Consider the formal descriptions of each set below. For each, write a short informal English description of each set.

1. $\{n | \exists_{m \in \mathbb{N}} : n = 2m\}$

   Set of all even natural numbers (0, 2, 4, ...).

2. $\{n | \exists_{m \in \mathbb{N}} : n = 2m \wedge \exists_{k \in \mathbb{N}} : n = 3k\}$

   Set of all natural numbers that are divisible by both 2 and 3 (6, 12, 18, ...) (i.e. divisible by 6).

3. $\{w | w \in \{0, 1\}^* \wedge w = w^R\}$

   Set of all palindromes over the alphabet $\{0, 1\}$ (0, 1, 00, 11, 000, 010, 101, 111, ...).

4. $\{n | n \in \mathbb{Z} \wedge n = n + 1\}$

   The empty set (there are no such integers n such that $n = n + 1$).

#3 Consider a square grid with length and width n. The bottom left corner is considered position (0, 0) and the upper right corner is position (n, n) (*Note that the first item in the tuple is the square along the horizontal axis and the second element is the index along the vertical axis). You can see an example grid below.

Our goal is to find all the unique ways a robot starting at cell (0, 0) can reach cell (n, n) by only moving up, down, left, right on the grid on each move. We would like you to do two things:

1. In your own words, argue why the given set (the set of possible paths to the goal) is infinite (as opposed to finite).

   Given that there are no bounds on our path length, the robot can move in cycles or loops without ever reaching the goal. For example, the robot could just move up and then down repeatedly for an infinite amount of steps, creating an infinite number of paths that take an infinite number of steps to reach the goal. Thus, the set of possible paths to the goal is infinite.

   Another way to say this, is given a finite set of paths, we can always add an extra loop in order to create an additional path. This can be done infinitely many times, thus the set of possible paths is infinite.

2. Show that the set of unique paths the robot can take to reach position (n, n) is countably infinite.

   If we place each possible movement in a list, we can create a bijection between the set of unique paths the robot can take to reach position (n, n) and the set of natural numbers. For example, we can list the movements as follows:

$$\Sigma = \{\text{up, down, left, right}\},$$
$$\Sigma^* = \{\text{all fininite paths using the moves up, down, left, right}\}$$

   Since Σ is a finite set, we know that Σ^* is countably infinite as we can create a bijection between the known countably infinite set $\{0, 1\}^*$ and Σ^*.

   The set of valid paths from $(0,0)$ to (n, n) is a subset of Σ^*, and a subset of a countably infinite set is either fininite or countably infinite. Since the set of valid paths is infinite, it must be countably infinite.

#4 Prove $F = \mathcal{P}(\mathbb{N})$ is uncountable.

Proof. **Cantor's Diagnolization Argument.** Assuming that $\mathcal{P}(\mathbb{N})$ is countable, then it's elements (subsets of $\mathbb{N}$) can be listed in a sequence $(S_1, S_2, S_3, \ldots)$. We can represent each of these elements as an infinite binary sequence, where the bit in the bitstring is 1 if the number is in the set and 0 if the number is not in the set. For example, if $S_1 = \{1, 3, 5\}$, then the binary representation of S_1 is $1010100000\ldots$

We can then construct a table of the binary representations of $S_1, S_2, S_3, \ldots$, and define a new subset $D \subseteq \mathbb{N}$ by flipping each bit on the diagonal of the table (i.e. the first bit of S_1, the second bit of S_2, etc.). For example, if the diagonal of the table is $1011\ldots$, then the binary representation of D is $0100\ldots$.

Since for every row we have a flipped bit, we can show that this new subset D is not in the list of subsets of binary representations of $\mathbb{N}$, which is a contradiction. Thus, $\mathcal{P}(\mathbb{N})$ is uncountable.

<div align="right">□</div>

#5 Suppose I build a new computing machine that can be programmed to recognize a lot of different functions! It is called the Flogrammable Device. In order to program this machine, you can type out your program on a tape, but this tape can only hold ten characters. Each character is from the alphabet $\Sigma = \{a, b, c, d, 0, 1\}$ and any combination of these 10 characters is a valid program. More precisely, a program is a String $P = p_1, p_2, \ldots, p_{10} | \forall_i p_i \in \Sigma$. You CANNOT have fewer than 10 characters or the code will not compile.

Now suppose that you read online that somehow, there are 100,000,000 important functions that need to be computed by the Flogrammable Device for it to cover all important functionality. Can we program each of the 100,000,000 functions on this machine? How do you know or not?

Since each program is a string of 10 characters, there are $6^{10} = 60,466,176$ possible programs that can be written on the Flogrammable Device. Since there are only 60,466,176 possible programs that can be written on the device, we cannot program all 100,000,000 functions on the device. Thus, we cannot program each of the 100,000,000 functions on the Flogrammable Device.