1. Use induction to prove the following summation formulas. Remember rules of logs. Specifically $\log(ab) = \log(a) + \log(b)$.

(a) $\sum_{i=1}^{n} a_i + b_i = \sum_{i=1}^{n} a_i + \sum_{i=1}^{n} b_i$

Proof.

Base case: $n = 1 \implies (a_1 + b_1) = (a_1) + (b_1)$

Inductive hypothesis: $n = k \implies \sum_{i=1}^{k} a_i + b_i = \sum_{i=1}^{k} a_i + \sum_{i=1}^{k} b_i$

Inductive case:

$$\begin{aligned}
\sum_{i=1}^{k+1} a_i + b_i &= \left(\sum_{i=1}^{k} a_i + b_i\right) + (a_{k+1} + b_{k+1}) \\
&= \left(\sum_{i=1}^{k} a_i + \sum_{i=1}^{k} b_i\right) + (a_{k+1} + b_{k+1}) \quad \text{by I.H.} \\
&= \left(\sum_{i=1}^{k} a_i\right) + a_{k+1} + \left(\sum_{i=1}^{k} b_i\right) + b_{k+1} \\
&= \sum_{i=1}^{k+1} a_i + \sum_{i=1}^{k+1} b_i
\end{aligned}$$

■

(b) $\sum_{i=1}^{n} ca_i = c\sum_{i=1}^{n} a_i$

Proof.

Base case: $n = 1 \implies ca_1 = ca_1$

Inductive hypothesis: $n = k \implies \sum_{i=1}^{k} ca_i = c\sum_{i=1}^{k} a_i$

Inductive case:

$$\begin{aligned}
\sum_{i=1}^{k+1} ca_i &= \left(\sum_{i=1}^{k} ca_i\right) + ca_{k+1} \\
&= \left(c\sum_{i=1}^{k} a_i\right) + ca_{k+1} \quad \text{by I.H.} \\
&= c\sum_{i=1}^{k+1} a_i
\end{aligned}$$

■

(c) $\sum_{i=1}^{n} \log(a_i) = \log\left(\prod_{i=1}^{n} a\right)$

Proof.

Base case: $n = 1 \implies \log(a_1) = \log(a_1)$

Inductive hypothesis: $n = k \implies \sum_{i=1}^{k} \log(a_i) = \log\left(\prod_{i=1}^{k} a_i\right)$

Inductive case:

$$\begin{aligned}
\sum_{i=1}^{k+1} \log(a_i) &= \sum_{i=1}^{k} \log(a_i) + \log(a_{k+1}) \\
&= \log\left(\prod_{i=1}^{k} a_i\right) + \log(a_{k+1}) && \text{by I.H.} \\
&= \log\left(\prod_{i=1}^{k+1} a_i\right) && \blacksquare
\end{aligned}$$

(d) $\sum_{i=1}^{n} 2i - 1 = n^2$

Proof.

Base case: $n = 1 \implies 2 - 1 = 1 = 1^2$

Inductive hypothesis: $n = k \implies \sum_{i=1}^{k} 2i - 1 = k^2$

Inductive case:

$$\begin{aligned}
\sum_{i=1}^{k+1} 2i - 1 &= \left(\sum_{i=1}^{k} 2i - 1\right) + 2(k+1) - 1 \\
&= k^2 + 2(k+1) - 1 && \text{by I.H.} \\
&= k^2 + 2k + 1 \\
&= (k+1)^2 && \blacksquare
\end{aligned}$$

(e) $\displaystyle\sum_{i=1}^{n} i^3 = \left(\frac{n(n+1)}{2}\right)^2$

Proof.

Base case: $n = 1 \implies \frac{1(1+1)}{2} = 1 = 1^3$

Inductive hypothesis: $n = k \implies \displaystyle\sum_{i=1}^{k} i^3 = \left(\frac{k(k+1)}{2}\right)^2$

Inductive case:

$$\begin{aligned}
\sum_{i=1}^{k+1} i^3 &= \sum_{i=1}^{k} i^3 + (k+1)^3 \\
&= \left(\frac{k(k+1)}{2}\right)^2 + (k+1)^3 && \text{by I.H.} \\
&= \left(\frac{k^2(k+1)^2}{4}\right) + (k+1)^3 \\
&= \frac{k^4 + 2k^3 + k^2}{4} + \frac{4k^3 + 12k^2 + 12k + 4}{4} \\
&= \frac{k^4 + 6k^3 + 13k^2 + 4}{4} \\
&= \frac{(k+1)^2(k+2)^2}{4} \\
&= \left(\frac{(k+1)(k+2)}{2}\right)^2 && \blacksquare
\end{aligned}$$

2. Prove that $n! > 2^n \quad \forall n > 4$

Proof.
Base case: $n = 5 \implies 5! > 2^5 \equiv 120 > 32$
Inductive hypothesis: $k \geq n \implies k! > 2^k$

Inductive case:

$$(k+1)! > 2^{k+1}$$
$$\implies k!(k+1) > 2^k(k+1) > 2^k(2) = 2^{k+1} \qquad \text{by I.H.}$$
$$\therefore \quad k!(k+1) > 2^{k+1}$$
■

3. A prime number p is a natural number greater than 1 where $1|p$ and $p|p$ and nothing else divides p. In class we showed that if n^2 is even, then n is even. Extend this to any prime number.

Prove that if p is prime and $p|n^2 \implies p|n$.
Equivalent Theorem: $p \nmid n \implies p \nmid n^2$.

Proof.

$$p \nmid n \implies n = pq + r, \quad \exists r > 0$$
$$\implies n^2 = nn = (pq+r)(pq+r) = p(pq^2 + 2qr) + r^2$$
$$\implies p \nmid p(pq^2 + 2qr) + r^2$$
■

4. Prove:

$$\sum_{i=1}^{n} ia^{i} = \frac{a-(n+1)a^{n+1}+na^{n+2}}{(a-1)^2}$$

Proof.

Base case: $n = 1 \implies$

$$a = \frac{a-2a^2+a^3}{(a-1)^2} = \frac{a(a^2-2a+1)}{a^2-2a+1} = a$$

Inductive hypothesis: $n = k \implies \sum_{i=1}^{k} ia^{i} = \frac{a-(k+1)a^{k+1}+ka^{k+2}}{(a-1)^2}$

Inductive case:

$$p\sum_{i=1}^{k+1} ia^{i} = \frac{a-(k+1)a^{k+1}+ka^{k+2}}{(a-1)^2} + (k+1)a^{k+1}$$

$$\frac{a-(k+1+1)a^{k+1+1}+(k+1)a^{k+1+2}}{(a-1)^2} = \frac{a-(k+1)a^{k+1}+ka^{k+2}}{(a-1)^2} + (k+1)a^{k+1} \text{ by I.H.}$$

$$\frac{a-(k+1+1)a^{k+1+1}+(k+1)a^{k+1+2}}{(a-1)^2} - \frac{a-(k+1)a^{k+1}+ka^{k+2}}{(a-1)^2} - (k+1)a^{k+1} = 0$$

$$(k+1)a^{k+1} - (k+1)a^{k+1} = 0 \qquad \text{by lots of algebra (used wolfram)}$$

$$0 = 0$$

■

5. Let's prove some theorems about cardinality. For each of the following equations, give a bijection between the two sets.

   Hint: you can (should) reuse a bijection you've already defined.

   (a) $|E| = |\mathbb{N}|, f : E \to \mathbb{N}, \quad f(e) = \frac{e}{2}$

   (b) $|\mathbb{N}| = |\mathbb{Z}|, g : \mathbb{N} \to \mathbb{Z}, \quad g(n) = \begin{cases} \frac{n}{2}, & 2|n \\ -\frac{n+1}{2}, & \text{else} \end{cases}$

   (c) $|\mathbb{N}| = |\mathbb{Q}^{+}|, h : \mathbb{N} \to \mathbb{Q}^{+}, \quad h(n) = \frac{n}{n+1}$

   (d) $|\mathbb{Z}| = |\mathbb{Q}|, k : \mathbb{Z} \to \mathbb{Q}, \quad k(z) = \begin{cases} h(n), & z > 0, \\ -h(n), & z < 0, \\ 0, & \text{else} \end{cases}$

   (e) $|E| = |\mathbb{Q}|, \lambda : E \to \mathbb{Q}, \quad \lambda(e) = k(g(f(e)))$

6. Prove that $\forall n, |\mathbb{N}| = |\mathbb{N}^n|$.

   You can assume that $d : \mathbb{N} \to \mathbb{N} \times \mathbb{N}$ is the diagonal bijection from class.

   Proof.
   Base case: $n = 1 \implies |\mathbb{N}| = |\mathbb{N}^1| = |\mathbb{N}|$
   Inductive hypothesis: $d_k : \mathbb{N} \to \mathbb{N}^k$ is a bijection.

   Inductive case:

$$\begin{aligned} d_{k+1} : \mathbb{N} \to \mathbb{N}^{k+1} &\implies |\mathbb{N}| = |\mathbb{N}^{k+1}| && \text{by I.H.} \\ &\therefore\ |\mathbb{N}| = |\mathbb{N}^k| = |\mathbb{N}^{k+1}| && \blacksquare \end{aligned}$$

7. Let's try to find a use for this infinity nonsense.

   In computer science it can be useful to look at problems as a language. A language is just a set of finite strings, so we can make a language describing π as:

$$L_\pi = \{"3", "3.1", "3.14", "3.141", \ldots\}$$

   So why do we care about languages? Well, we can phrase all of our problems in CS as different languages. For example $L_{factor} = \{"6 = 2 \cdot 3", "12 = 2 \cdot 2 \cdot 3", \ldots\}$ is the language of numbers and their factors. We can make the language of graphs and their shortest paths, the languages of lists and their sorting. Really we can make a language for any problem.

   A language is decidable if there is a program that can (eventually) produce any string in that language. We want to prove that there is at least 1 undecidable language. That is, there is a problem that can't be solved by a program.

   (a) Show that there are countably many programs we can write.

   Need to show: there exists an injective function from $S \to \mathbb{N}$

   Let f be a function that takes program and returns a binary sequence that represents the instructions of that program and vice versa.

   If two binary sequences are equal, then they must produce the same program, implying the function is injective ■

   (b) Show that are uncountably many languages.

   Since π can be described above as L_π, then two unique strings could easily map the same output. Thus, it cannot be an injective function and $|L_\pi| > |N|$ ■

8. Show why the following two lemmas are actually invalid.
   Note: the theorem is actually true despite this (haha)

   Hint: part 1 is a problem with induction and part 2 is a problem with infinity.

   Lemma 0.1. *There are infinitely many green vegetables*.

   Proof.

   Base case: There exists a head of lettuce.
   Inductive hypothesis: there is a set of k green vegetables.
   ↑ ERROR: the inductive hypothesis must say something about the set of vegetables, not just that the set exists. This is **not a valid hypothesis**.

   Inductive case: Suppose, by the inductive hypothesis, there are k green vegetables.

   Remember that head of lettuce from from the base case? Let's add that into our set of k vegetables. Now we have a set of $k + 1$ green vegetables.

   Therefore we must have a set of n vegetables for any n, which means we have infinitely many green vegetables. ■

   Lemma 0.2. *Every green edible thing is a vegetable*.

   Proof.

   Let's consider the set of all possible green edible things G. By the previous lemma G is clearly an infinite set. I'll sort the set putting all of the green vegetables at the front. Now we want to show that there are ONLY vegetables in this set.
   ↑ ERROR: (1) G is clearly not an infinite set. (2) You can't sort an infinite set, therefore, the following hypothesis is an impossible claim.

   Base case: the first element is that dang head of lettuce from the last proof.
   Inductive hypothesis: The first k elements of G are vegetables.
   Inductive case: Assume, by the inductive hypothesis, that the first k elements are vegetables. Then the next element must also be a vegetable. If it wasn't then there would only be a finite number of vegetables, but by the last theorem this is impossible.

   Therefore, the first $k + 1$ elements of G must be vegetables. Thus, for all n, the first n elements of G are vegetables. Now, since G is a countably infinite set, and we've shown that each position in the set is a vegetable, it must be the case that the set contains only vegetables.

   Theorem 0.2.1: *Mountain Dew is a vegetable*.

   Proof.

   It's green and edible. From the previous two lemmas, it's clearly a vegetable. ■