Induction III

1 Two Puzzles

Here are two challenging puzzles.

1.1 The 9-Number Puzzle

The numbers $1, 2, \ldots, 9$ are arranged in a 3×3 grid as shown below:

1	2	3
4	5	6
7	8	9

You can rearrange the numbers by *rotating* rows and columns. For example, rotating the first row to the right gives:

1	2	3
4	5	6
7	8	9

$\longrightarrow$

3	1	2
4	5	6
7	8	9

Notice that 1 and 2 both moved right one position and the rightmost number, 3, jumped back to the left. Similarly, if we now rotate the first column downward, then 3 and 4 both move down one position and the bottom number, 7, jumps back up to the top:

3	1	2
4	5	6
7	8	9

$\longrightarrow$

7	1	2
3	5	6
4	8	9

Can you find a sequence of moves that *transposes* the original configuration?

1	2	3
4	5	6
7	8	9

$\longrightarrow \quad \ldots \quad ? \quad \ldots \quad \longrightarrow$

1	4	7
2	5	8
3	6	9

1.2 The Temple of Forever

Each monk entering the Temple of Forever is given a bowl with 15 red beads and 12 green beads. Each time the Gong of Time rings, the monk must do one of two things:

1. If he has at least 3 red beads in his bowl, then he may remove 3 red beads and add 2 green beads.
2. He may replace every bead in his bowl with a bead of the opposite color.

For example, at the first ring of the Gong of Time, the monk might replace every bead with one of the opposite color, giving him 12 red and 15 green. Then, at the second ring, he might remove 3 red and add 2 green, leaving 9 red and 17 green.

A monk may leave the Temple of Forever only when he has exactly 5 red beads and 5 green beads in his bowl. Can you find a way to escape?

2 Using Induction to Analyze a Process

An important application of induction is proving that a system never enters some undesirable state. For example, we might want to prove that a file system is never corrupted, records in a data structure are always rapidly retrievable, or a communication protocol never deadlocks. No new mathematical techniques are required to use induction for such purposes. But you'll need to think about induction somewhat differently. We'll use the 9-Number Puzzle and the Temple of Forever as illustrations.

2.1 Induction on Time

Some frustrating experiments suggest that there is *no way* to escape from the Temple of Forever or solve the 9-Number Puzzle. But how could we hope to prove these conclusions using induction? Remember that induction establishes that some predicate $P(n)$ is true for all $n \in \mathbb{N}$. For the Temple problem, should we use induction on the number of red beads? Green beads? The total? Worse, the 9-Number Puzzle doesn't seem to involve a natural-valued variable at all!

The common solution when analyzing a system is to use induction on *time*; that is, we'll use induction to prove that some predicate $P(n)$ is true for every $n \geq 0$ where n is the number of rotations, gong rings, moves, hours, steps, or whatever.

Unfortunately, a naive approach still doesn't work. Let's try such an argument and see where we get stuck.

Theorem. *No one leaves the Temple of Forever.*

Proof. We use induction. Let $P(n)$ be the proposition, "After n gong rings, the number of red beads in the monk's bowl is not equal to the number of green beads."

Base case. Initially, there are 15 red beads and 12 green beads, so $P(0)$ is true.

Inductive step. We must show that $P(n)$ implies $P(n+1)$ for all $n \geq 0$. So assume that after n gong rings the number of red beads in the monk's bowl is not equal to the number of green beads. Then after $n+1$ gong rings... $\times$

We're stuck! If we assume that $P(n)$ is true, the monk might have, say, 8 red beads and 3 green beads after n gong rings. But then removing 3 red and adding 2 green leaves him with 5 red and 5 green, making $P(n+1)$ false! In other words, we can't hope to prove $P(n)$ implies $P(n+1)$ in the inductive step— not because we aren't sufficiently clever, but because it's just not true!

So we must be clever-*er*.

2.2 Finding an Induction Hypothesis

The key to proving that a system can never reach a "bad" state is chosing the right induction hypothesis. In particular, the induction hypothesis should describe a property that is:

1. True at the start.
2. ***Invariant***, meaning that if the system has the property *before* a move, then it must also have the property *after* the move.
3. False in the "bad" state.

Intuitively, we're looking for a property that the system has initially and can never lose. This means any state of the system lacking that property is unreachable.

Let's check a couple properties against these criteria.

The monk always has at least one bead. This is true initially since the monk starts with a bowlful of beads. Furthermore, this property is invariant. Suppose the the monk has at least one bead.

- If the monk removes 3 red beads and adds 2 green, then he is left with at least the 2 green. (Remember this operation is only allowed if the monk *has* at least 3 red.)
- If the monk has at least one bead before swapping colors, then he has at least one bead after doing so; in fact, chaning colors does not affect the number of beads at all.

The problem is that this property also holds in the state we're trying to rule out, where the monk has 5 red and 5 green. So this property does not meet all our criteria.

The monk has an unequal number of red and green beads. This property does hold at this start, and does not hold in the "bad" state where the monk has 5 red and 5 green. However, this property is not preserved by every move. For example, if the monk has 13 red and 8 green, then after the next gong he could have 10 red and 10 green. In other words, the property is not invariant.

A good way to find an invariant is to list a lot of states and look for a distinctive feature they have in common. For example, here are some states the monk can reach:

$$\begin{array}{ccccccc} (15,12) & \to & (12,15) & \to & (9,17) & \to & (17,9) \\ \downarrow & & & & & & \\ (12,14) & \to & (14,12) & \to & (11,14) & \to & (14,11) \\ \downarrow & & & & \downarrow & & \downarrow \\ (9,16) & \to & (16,9) & & (8,16) & & (11,13) \end{array}$$

Here the pair (r, g) denotes the state where the monk has r red beads and g green beads. Continuing in this way, you might notice that the difference $r - g$ only takes on certain values: 2, -2, 3, -3, 7, -7, 8, -8, etc. *In particular, the number of red beads minus the number of green beads is always of the form $5k+2$ or $5k+3$ where k is an integer.* The rules for the Temple provide an explanation: adding 3 red and removing 2 green changes the difference by 5. And swapping colors negates the difference. Furthermore, this property holds at the start (since $15 - 12 = 5 \cdot 0 + 3$) and does not hold in the state we're trying to prove unreachable (since $5 - 5 = 0$ is not of the form $5k + 2$ or $5k + 3$). This is exactly the sort of property we need, so we're ready for a proof!

Theorem 1. *No one leaves the Temple of Forever.*

Proof. We use induction on the number of gong rings. Let $P(n)$ be the proposition that after n rings, the number of red beads in the monk's bowl minus the number of green beads is equal to $5k + 2$ or $5k + 3$ for some integer k.

Base case: $P(0)$ is true because initially (after zero rings) the number of red beads minus the number of green beads is $15 - 12 = 5 \cdot 0 + 3$.

Inductive step: Now assume that $P(n)$ holds after n gong rings, where $n \geq 0$. Let r denote the number of red beads in the monk's bowl, and let g denote the number of green beads. In these terms, we are assuming that $r - g$ is equal to $5k + 2$ or $5k + 3$ for some integer k. After $n + 1$ gong rings, there are two cases to consider, depending on the monk's action:

1. If $r \geq 3$, then the monk may have replaced 3 red beads with 2 green beads. Thus, the number of red beads minus the number of green becomes:

   $$(r - 3) - (g + 2) = (r - g) - 5$$

   This is equal to either $5(k - 1) + 2$ or $5(k - 1) + 3$, so $P(n + 1)$ is true.

2. Alternatively, the monk may have exchanged every red bead for a green bead and vice versa. In this case, the number of reds minus the number of greens becomes $g - r$. If $r - g = 5k + 3$, then $g - r = 5(-k) - 3 = 5(-k - 1) + 2$. If $r - g = 5k + 2$, then $g - r = 5(-k) - 2 = 5(-k - 1) + 3$. Thus, $P(n + 1)$ is again true.

Therefore, $P(n)$ implies $P(n + 1)$ for all $n \geq 0$.

By the induction principle, $P(n)$ is true for all $n \geq 0$. Since the number of red beads minus the number of greens is always of the form $5k + 2$ or $5k + 3$ and the difference required to leave the temple does not match either form, no monk can ever leave the Temple of Forever. □

Many proofs that a system can not reach a "bad" state share several features with this one:

- The induction hypothesis holds initially and is preserved by every move, but does not hold in the "bad state".
- The proof uses induction on time, as measured by gong rings, operations, or whatever.
- In the inductive step, there is one case for every possible operation.

In fact, the proof that the 9-Number Puzzle is unsolvable has the same format.

3 The 9-Number Puzzle

We'll prove the 9-Number Puzzle insoluble using a property that probably would not occur to you immediately, but comes up all the time in puzzles and protocols involving arrangements of objects.

3.1 Permutations and Inversions

A ***permutation*** of the numbers 1 thorugh 9 is a sequence containing each digit exactly once. Placing the rows of the puzzle side-by-side gives a permutation of the numbers 1 to 9. For example, the original configuration corresponds to a permutation as shown below:

1	2	3
4	5	6
7	8	9

$\longrightarrow \quad 1\,2\,3\,4\,5\,6\,7\,8\,9$

An ***inversion*** is a pair of terms in a permutation that are in reverse order. For example, the original permutation has *zero* inversions; 1 precedes 2, 1 precedes 3, 2 precedes 3, 1 precedes 4, and so forth. But now suppose we rotate the top row to the right as before:

3	1	2
4	5	6
7	8	9

$\longrightarrow \quad 3\,1\,2\,4\,5\,6\,7\,8\,9$

Now there are *two* inversions: 3 and 1 are out of order and so are 3 and 2. Suppose we rotate the first column downward:

7	1	2
3	5	6
4	8	9

$\longrightarrow \quad 7\,1\,2\,3\,5\,6\,4\,8\,9$

The resulting permutation has *eight* inversions:

$$(7,1)\quad (7,2)\quad (7,3)\quad (7,5)$$
$$(7,6)\quad (7,4)\quad (5,4)\quad (6,4)$$

At this point, one might guess that *the number of inversions is always even*. Let's check that this property does not hold for the transpose configuration, which we'd like to prove unreachable:

1	4	7
2	5	8
3	6	9

$\longrightarrow \quad 1\,4\,7\,2\,5\,8\,3\,6\,9$

Sure enough, here there are *nine* inversions, which is an odd number:

$$(4,2)\quad (4,3)\quad (7,2)\quad (7,4)\quad (7,3)$$
$$(7,6)\quad (5,3)\quad (8,3)\quad (8,6)$$

3.2 The 9-Number Puzzle is Impossible

We're now ready to prove that the 9-Number Puzzle is unsolvable. The proof relies on some preliminary facts about inversions. This is exactly the sort of situation where lemmas can make an argument more clear.

The ***parity*** of an integer refers to whether the number is even or odd. For example, 7 has odd parity, 4 has even parity, and 0 has even parity.

Lemma 2. *Swapping two terms in a permutation changes the parity of the number of inversions.*

Proof. Take an arbitrary permutation:

$$\text{——}\; x\; b_1 \ldots b_k\; y \;\text{——}$$

The horizontal lines indicate sequences of elements. Swapping x and y gives the permutation:

$$\text{——} \; y \; b_1 \ldots b_k \; x \; \text{——}$$

This reverses the order of $2k+1$ pairs: x and y, x and each b_i, and y and each b_i. Effectively, this flips the parity $2k+1$ times which is equivalent to flipping the parity once. □

Lemma 3. *Rotating three terms in a permutation preserves the parity of the number of inversions.*

Proof. Take an arbitrary permutation:

$$\text{——} \; x \; \text{——} \; y \; \text{——} \; z \; \text{——}$$

A forward rotation is equivalent to swapping y and z and then swapping x and z:

$$\text{——} \; z \; \text{——} \; x \; \text{——} \; y \; \text{——}$$

Similarly, a reverse rotation is equivalent to swapping x and y and then x and z:

$$\text{——} \; y \; \text{——} \; z \; \text{——} \; x \; \text{——}$$

In any case, two swaps flip the parity twice, which leaves the original parity unchanged. □

Theorem 4. *No sequence of moves transforms the original configuration of the 9-Number Puzzle into the transpose configuration.*

Proof. We use induction. Let $P(n)$ be the proposition that after n steps the number of inverted digits is *even*.

Base case. After 0 steps, the puzzle is in the original configuration. There are zero inversions, so $P(0)$ is true.

Inductive step. Suppose that after n steps, the puzzle has an even number of inversions. By Lemma 3, rotating three digits preserves the parity of the number of inversions. Thus, the puzzle has an even number of inversions after $n+1$ steps as well.

By the principle of induction, $P(n)$ is true for all $n \geq 0$; that is, the number of inversions is even after any sequence of moves. The transpose configuration is unreachable since it has an odd number of inversions. □

A few wrap-up notes, First, notice that Lemma 3 holds when *any* three digits are rotated. Thus, even if we allowed rotations along the long diagonals of the 9-Number Puzzle, there would *still* be no way to reach the transpose configuration. Second, similar arguments about permutations and inversions explain why certain states are unreachable in many other puzzles. For example, you may have observed that there is no way to flip a single edge of Rubik's Cube. Finally, you might be discouraged that this technique only helps prove that puzzles are *not* solvable. This seems rather negative. But remember in that in the context of a file system or communication protocol, proving that the system never enters a corrupted or deadlocked state is a very *good* thing!

4 Common Induction Mistakes

There are several potholes that students commonly fall into while trying to write induction proofs. Some are simple, some are quite subtle. Collectively, these traps cost 6.042 students *thousands* of points every term. Here are the top grade-gutters and how to avoid them. (Alternatively, if you mash all these blunders into a single proof, you might be able to drive your TA bananas.)

4.1 The Misplaced Quantifier

Let's start with a simple, relatively minor gotcha. Can you spot the problem?

Theorem 5. *For all $n \geq 0$:*

$$1^2 + 2^2 + \ldots + n^2 = \frac{n(2n+1)(n+1)}{6}$$

Proof. We use induction. Let $P(n)$ be the proposition, "For all $n \geq 0$, $1^2 + 2^2 + \ldots + n^2 = n(2n+1)(n+1)/6$".

Base case. Etc. $\times$

In general, an induction hypothesis is a predicate $P(n)$, which is a statement that is true or false depending on the value of n. The goal of an induction proof is to prove that $P(n)$ is true for all $n \geq 0$. A valid induction hypothesis in this case would be:

$$P(n) = \text{“}1^2 + 2^2 + \ldots + n^2 = \frac{n(2n+1)(n+1)}{6}\text{”}$$

In the erroneous proof above, the predicate $P(n)$ *itself* asserts that an equation holds for all $n \geq 0$. This makes no sense. The "For all $n \geq 0$" bit should *not* be part of the induction hypothesis.

4.2 Misusing a Predicate as a Numerical Function

Here's another classic. This one is a pretty major error.

Theorem 6. *For all $n \geq 0$:*

$$1^2 + 2^2 + \ldots + n^2 = \frac{n(2n+1)(n+1)}{6}$$

Proof. We use induction. Let $P(n)$ be "$1^2 + 2^2 + \ldots + n^2 = n(2n+1)(n+1)/6$".

Base case. $P(0) = \frac{0(2\cdot 0+1)(0+1)}{6} = 0$

Inductive Step.

$$\begin{aligned} P(n) + (n+1)^2 &= \frac{n(2n+1)(n+1)}{6} + (n^1)^2 \\ &= \frac{(n+1)(2(n+1)+1)(n+2)}{6} \\ &= P(n+1) \end{aligned}$$

×

Remember, an induction hypothesis is a predicate $P(n)$, which is a statement that is true or false depending on the value of n. In particular, $P(n)$ has *no a numerical value*. Adding $P(n)$ is like trying to divide by a pomegranate. It makes no sense.

4.3 Too Few Base Cases

The ***Fibonacci numbers*** $F_0, F_1, F_2, \ldots$ are defined recursively as follows:

$$F_n = \begin{cases} 0 & \text{when } n = 0 \\ 1 & \text{when } n = 1 \\ F_{n-1} + F_{n-2} & \text{when } n \geq 2 \end{cases}$$

Thus, the first few Fibonacci numbers are 0, 1, 1, 2, 3, 5, 8, ... Each number is the sum of its two predecessors, except for the first two.

We'll use Fibonacci numbers to demonstrate a error that usually comes up in connection with strong induction proofs.

False Claim 7. *All Fibonacci numbers are even.*

Proof. We use strong induction. Let $P(n)$ be the proposition that F_n is even.

Base case. $F_0 = 0$ is even, so $P(0)$ is true.

Inductive step. Assume $P(0), \ldots, P(n-1)$ to prove $P(n)$. Now

$$F_n = F_{n-1} + F_{n-2}$$

and F_{n-1} and F_{n-2} are both even by assumptions $P(n-1)$ and $P(n-2)$, so F_n is also even.

By induction, all Fibonacci numbers are even. ×

The problem is that too few base cases are considered. This immediately raises a larger question: "How many base cases must I consider?" The answer goes back to the strong induction axiom. In order to prove $P(n)$ for all $n \geq 0$, you must prove *all* of the following:

☐	1.	$P(0)$
☐	2.	$P(0)$ implies $P(1)$
☐	3.	$P(0)$ and $P(1)$ imply $P(2)$
☐	4.	$P(0), P(1),$ and $P(2)$ imply $P(3)$
☐	5.	$P(0), P(1), P(2)$ and $P(3)$ imply $P(4)$
...	...	etc.

You can regard this as a scorecard for strong induction proofs; if you can't check off every box, the proof is bogus. For example, in the "proof" above, we established the first statement under the *base case* heading. The argument under the *inductive step* heading established statements 3, 4, 5, etc. (This argument does not work for $n = 1$, because it relies on the assumption $P(n-2)$.) So here's the scorecard:

☒	1.	$P(0)$
☐	2.	$P(0)$ implies $P(1)$
☒	3.	$P(0)$ and $P(1)$ imply $P(2)$
☒	4.	$P(0), P(1),$ and $P(2)$ imply $P(3)$
☒	5.	$P(0), P(1), P(2)$ and $P(3)$ imply $P(4)$
...	...	etc.

In order to complete the proof, we would need to show that $P(0)$ implies $P(1)$. But we can not in this case, since $P(1)$ is actually false; the Fibonacci number $F_1 = 1$ is odd.

4.4 Choosing the Wrong Induction Variable

Many problems involve several different natural-valued random variables. For example:

Theorem 8. *For all integers $k \geq 0$ and $r \geq 2$:*

$$1 + r + r^2 + \ldots + r^k = \frac{1 - r^{k+1}}{1 - r}$$

We could try an induction argument based on the variable r or based on the variable k. One choice (k) leads to prompt success and the other (r) leads to disaster. How are you to know which is which?

There is no general answer, though we'll try to provide guidance from time to time. For example, when proving a property of a system that changes in discrete time steps, use induction on the number of steps. Your best course is to avoid fixating exclusively on one variable; if the proof doesn't work out, try induction on another variable.

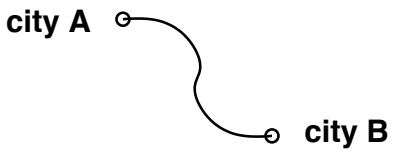
4.5 Build-Up Error

This last error is both subtle and common. An *autocity* is a city with a road to some other town.

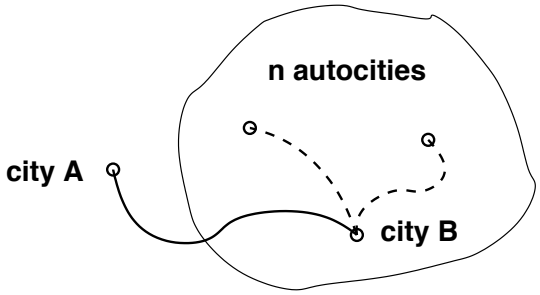
Claim 9. *You can drive between every pair of autocities in our world.*

Proof. The proof is by induction. Let $P(n)$ be the proposition that in all possible worlds with n autocities, you can drive between every pair of autocities.

Base case. Since a world with $n = 1$ autocities can not exist, we begin by proving $P(2)$. In this case, only one configuration is possible, and the claim clearly holds:



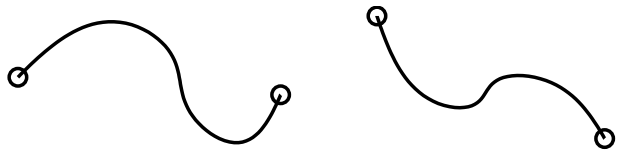
Inductive step. Next, we show that $P(n)$ implies $P(n+1)$ for all $n \geq 2$. As usual, we assume that $P(n)$ is true and show that $P(n+1)$ logically follows. By the assumption $P(n)$, you can drive between every pair of autocities in a world with n autocities. Now we add one more autocity (call it "city A") to form a world with $n+1$ autocities:



All that remains is to prove that you can drive from city A to any other city. Since A is an autocity, you can drive from A to at least one of the other n autocities (call that one "city B"). Then, from city B, you can drive to any of the others, by our assumption $P(n)$. This shows that $P(n+1)$ is true.

By induction, $P(n)$ is true for all $n \geq 2$. Thus, in particular, you can drive between any pair of autocities in our world. □

The error is in the inductive step. We showed that the induction hypothesis holds for every world with $n+1$ autocities *which can be built-up by adding one autocity to a world with* n *autocities*, you can drive between any two autocities. However, *not all worlds* with $n+1$ autocities can be built up by adding one more autocity to a world with n autocities. Here is an example:



There is no way to construct this world by adding 1 more autocity to a world with 3 autocities. Therefore, we have *not* shown that $P(n)$ implies $P(n+1)$ and the induction argument is broken.

This blunder of assuming an arbitrary configuration of $n+1$ objects can be built up from a good configuration of n objects in some particular way is known as "build-up error."