

Mathematical Induction

Good Explanations

Throughout this course, we've stressed that **good explanations are at least as important as right answers**. In many ways, explanations are *more* important, because explanations can be modified and generalized to solve other related problems. Answer are just... answers.

We've also stressed --- over and over again --- the following true fact about explanations:

A list of examples does not explain why a statement is always true.

If we believe that the statement *The binary representation of an even number always ends in a zero* is true, it's not enough to demonstrate that 2 is `10`, 4 is `100`, and 6 is `110`. All three of those are correct examples. But the statement doesn't say that the binary representation of an even number is *sometimes* true. It says **always**, and three examples doesn't prove "always". Three million examples wouldn't do it either.

To give an explanation of why a statement is **always** true, examples can be important to help us think about the situation; but in the end, we need an **explanation that does not depend on examples** and which **uses the core concepts involved in the problem to explain things**.

Here, for example, would be an explanation for why the binary of an even number always ends in a zero:

Let n be any even number. Then if we convert n to binary, the rightmost bit will be the remainder left over when we divide n by 2. Since n is even, it's evenly divisible by 2 with no remainder. Therefore that rightmost bit must be zero.

Here's another, which takes a different approach:

The statement we are explaining says "If n is even, then its binary representation ends in a zero". The contrapositive of this statement would say: "If the binary of n ends in 1, then n is odd." The contrapositive is logically equivalent to the original, so if we can explain why *it* is true, then the original is also explained. So: Let n be a number whose binary ends in 1. Then regardless what the other bits are, they represent multiples of 2. Since the ending bit is 1, the number is a multiple of 2, plus 1. This makes it an odd number because all odd numbers have this form --- they are multiples of 2, plus 1.

You might have questions about why some of the individual steps are true. But you can't deny that each step follows logically from the one before it, and no specific examples are ever used. That makes these correct and fully generalized explanations that explain why the statement is **always** true.

These kinds of general explanations, that explain why a statement is always true, are sometimes called **proofs**. We'll use that term a lot. But remember, **a proof is just a clear, convincing explanation of why a statement is always true.**

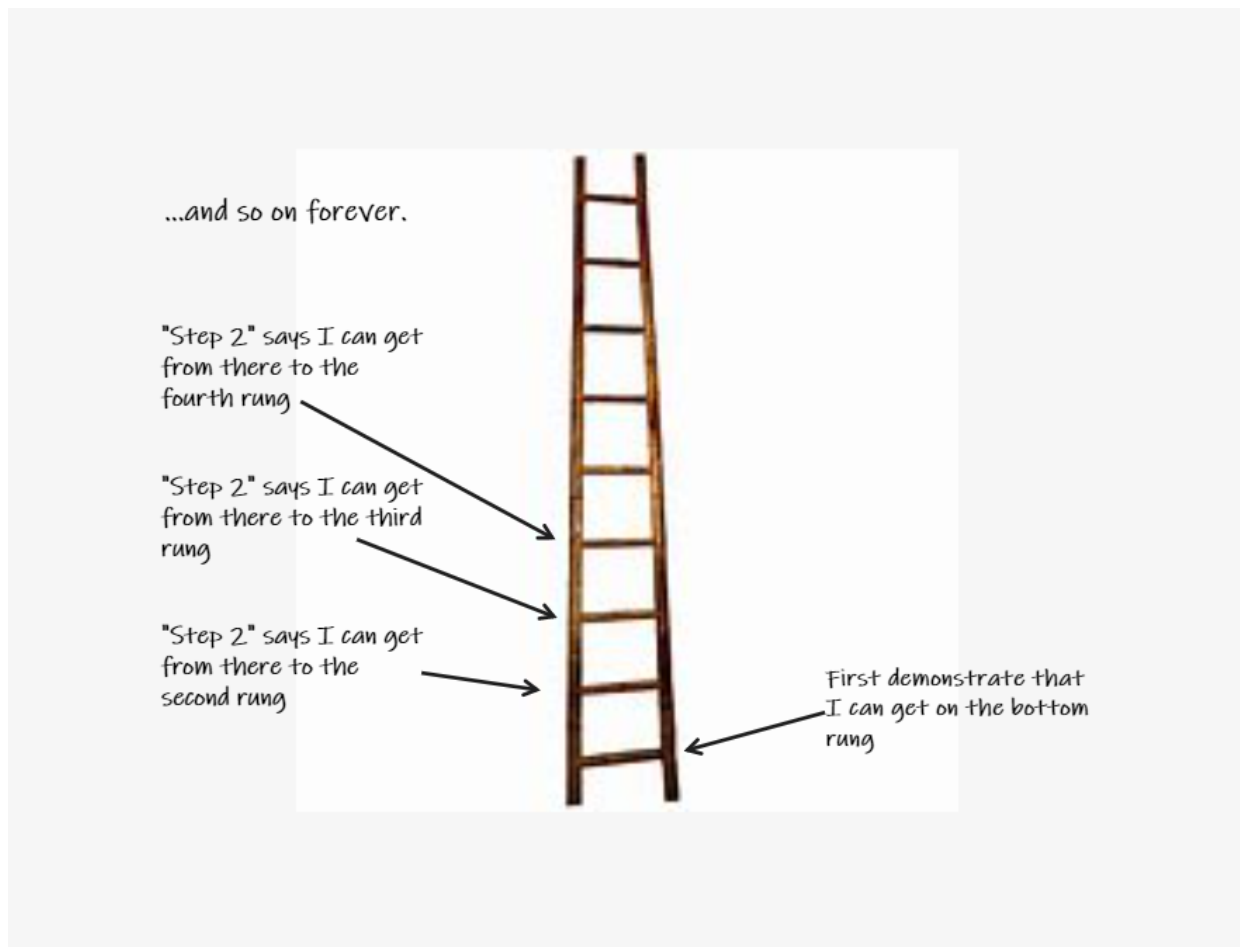
What is mathematical induction?

Mathematical induction is a particular method of proof that is typically used in explanations of statements involving recursive problems. These can include certain kinds of counting problems; problems about sums or sequences; problems about recurrence relations; and more. Whenever a statement is being made about something recursive, the proof will usually use mathematical induction.

Imagine that I have a very tall ladder, and I want to prove to you that I can reach the top of the ladder, without actually getting the ladder out and simply climbing it. (Because I'm lazy.) How might I convince you? I might do this in two steps:

- **Step 1: I would physically step onto just the lowest rung of the ladder**, to show you that I am physically capable of getting on the ladder in the first place. (I'm not *that* lazy.)
- **Step 2: I would explain to you the process I would go through to get from one rung of the ladder to the next.** "You hold on to the sides, lift up your right foot and place it on the next rung, then push up with your other foot." If that explanation of the climbing process is valid, then you will agree that I know how to get from any one rung of the ladder to any other.

Those two steps put together prove that I can climb as high as I want on the ladder. Because I've shown you in Step 1 that I can get onto the lowest rung. Then the Step 2 implies that I can get from the lowest rung to the one above it. Then Step 2, used again, implies that I can get from *that* rung to the one above *it*. Then Step 2 implies that I can get from *that* rung to the one above *it*; and so on, forever.



When is mathematical induction used?

Mathematical induction involves the use of our old friend, the **predicate**. Remember a predicate is like a logical statement, but it has a variable in it, and the value of the predicate could be either True or False depending on the variable. A predicate, in other words, is a *function* whose codomain is the set {True, False}. You plug in values of the variable, and you get either True or False out.

More specifically, mathematical induction involves predicates whose *domains* (the input data) are natural numbers or some subset of the natural numbers. Here are three examples:

- $P(n)$ which is the statement " n is an even number".
- $Q(n)$ which is the statement " $n^2 \geq 0$ ".
- $R(n)$ which is the statement " $n^3 + 2n$ is a multiple of 3".

The first predicate $P(n)$ is clearly True for some inputs but False for others. The second one is clearly always true, since squaring a nonnegative number always produces a nonnegative number. What about the third one? Is that predicate *always* true or just *sometimes* true? We can explore the statement by generating some values (here, we're using a [Python list comprehension](#)):

```
> [n**3 + 2*n for n in range(10)]

# output
> [0, 3, 12, 33, 72, 135, 228, 357, 528, 747]
```

We can check that all ten of those values are indeed divisible by 3. (For example, apply the function $x \% 3$ to the list.) If you go back and generate a longer list, you'll see that all the examples generated will be divisible by 3.

Does that mean that the predicate $R(n)$ is always true? **NO!** Because that's only a list of ten examples and remember, examples do not prove something is always true. However, it might make you think that, *maybe*, the predicate is always true. So this is when we want to start thinking about a *proof*.

This is when mathematical induction is used: **When trying to prove that a predicate involving natural numbers is always true.**

How does mathematical induction work?

There are three distinct and related steps in a proof by mathematical induction. Keep the metaphor of the ladder in mind.

Mathematical Induction

- **Step 1: Prove that the predicate is true in the smallest possible case.** For example, if we are proving that "For all natural numbers n , the number $n^3 + 2n$ is a multiple of 3" then the smallest possible case is where $n = 0$. This is called the **base case**.
- **Step 2: Assume that the predicate is true for some unspecified value of the input, say k .** That is, let k be some random value of the input and assume that $P(k)$ is true. This assumption is called the **induction hypothesis**.
- **Step 3: Explain why the predicate is true for the value of the input just after k .** That is, having assumed that $P(k)$ is true, explain why $P(k + 1)$ is true.

What does this have to do with the ladder?

- Demonstrating that the predicate is true in the base case, is like me physically stepping onto the bottom rung of the ladder to show you I am capable of getting on the ladder at all.
- Assuming the induction hypothesis --- assuming $P(k)$ is true for some value k of the input --- is like saying, "Suppose I'm standing on the k th rung of the ladder."
- Then proving that $P(k + 1)$ is true, is like saying, "...and now I'll show you how to get to the next rung."

And just like the ladder, the combination of those three parts proves that the predicate is true *for all* values of the input. I can get to *any* rung on the ladder I want if I can get on the bottom rung and then show how to get from any one rung to the next.

Didn't you say something about recursion earlier? Yes, I did, and it shows up in between Steps 2 and 3 in this process. In an induction argument, we need to use the fact that $P(k)$ is true to prove $P(k + 1)$ is true. **That's recursion!** I'm building the truth of $P(k + 1)$ from the truth of the predicate in a smaller case.

The "framework" for a mathematical induction proof

The three steps above are what we'll often call the **framework** for a proof by mathematical induction. To "set up the framework" for a proof by mathematical induction, we start with a predicate P and we claim that P is true for some subset of the natural numbers (possibly all of \mathbb{N}). Then we do three things:

1. **Prove the base case:** Demonstrate directly that P is true in the smallest possible case.
2. **State the induction hypothesis:** Write out the assumption being made in Step 2 above, in clear language and in the context of the problem. We usually start by saying: **For some natural number k , assume that $P(k)$ is true**, and then rephrase the last part (" $P(k)$ is true") using the context of the problem.
3. **State precisely what you need to prove and then outline how we might prove it.** State $P(k + 1)$ says in terms of the problem. Then give an overview of how we'd explain why $P(k + 1)$ is true having assumed $P(k)$ is true.

Examples

Example 1: Multiples of 3

For all natural numbers n , $n^3 + 2n$ is a multiple of 3.

The predicate here is $P(n) : n^3 + 2n$ is a multiple of 3.. Here is the set-up framework:

- **Base case:** The claim is that $P(n)$ is true for all natural numbers, so the smallest of those is $n = 0$. So here, we want to directly demonstrate that $P(0)$ is true. Well, $P(0)$ is the statement that $0^3 + 2(0)$ is a multiple of 3. The expression $0^3 + 2(0)$ is equal to 0, and this is a multiple of 3, namely 3×0 . So that shows $P(0)$ is true, and therefore we've established the base case.
- **Induction hypothesis:** Boilerplate for this is: **Let k be some natural number. Then assume that $P(k)$ is true.** That's not wrong but we'd like a little more detail in the second part. So here is a better induction hypothesis:

Let k be some natural number. Assume that $k^3 + 2k$ is a multiple of 3.

This is "better" because the second part has been phrased *in the context of the problem* --- " $k^3 + 2k$ is a multiple of 3" and not just the bland " $P(k)$ is true". It just replaces the literal $P(k)$ with what $P(k)$ says.

- **Proof step:** Now we outline/brainstorm how we'd prove $P(k + 1)$ is true assuming $P(k)$ is true. This takes some playing around and some creativity, but here's how that might go here:

We want to show that $P(k+1)$ is true, that is, $(k+1)^3 + 2(k+1)$ is a multiple of 3. Probably I would multiply out the terms in this expression, and look for places where $k^3 + 2k$ appears. I've assumed that this is a multiple of 3, so whenever I see it I can replace it with a multiple of 3.

The framework is not asking for a full-blown proof of the statement, just an outline. But: Here is what a full-blown proof *would* look like.

To Prove: For all natural numbers n , $n^3 + 2n$ is a multiple of 3.

Proof: First, to prove the base case, if $n = 0$ then we have $0^3 + 2(0)$ which equals 0. This is a multiple of 3 because $0 = 3 \times 0$. Therefore the base case is proven.

Now assume that for some natural number k , $k^3 + 2k$ is a multiple of 3. We want to show that $(k+1)^3 + 2(k+1)$ is a multiple of 3. Expand this out to get:

$$\begin{aligned}(k+1)^3 + 2(k+1) &= (k^3 + 3k^2 + 3k + 1) + (2k + 2) \\ &= (k^3 + 2k) + 3k^2 + 3k + 3\end{aligned}$$

Now we already assumed that $k^3 + 2k$ was a multiple of 3, and clearly the last three terms are multiples of 3 because 3 is a factor on each. Therefore $(k+1)^3 + 2(k+1)$ is a sum of multiples of 3, which makes the entire expression a multiple of 3.

That ends the proof. 👍

Notice a few things:

- Notice that when we get to the proof step, we are using $P(k+1)$. This always involves plugging in $k+1$ in for the variable that is in the predicate.
- Not all of the algebra that I *could* have done in the first line of the equations above, was actually done. I didn't simplify $(k+1)^3 + 2(k+1)$ all the way, because I knew from my outline that I want to look for $k^3 + 2k$. I saw it there, so instead of simplifying *everything* I moved the $2k$ up next to the k^3 and then simplified the rest.
- Notice the recursion: We used something known about $k^3 + 2k$ to prove something about $(k+1)^3 + 2(k+1)$. **If you are not using the induction hypothesis in a proof by induction, something is not right.**
- Were you expecting more at the end? There isn't anything left to do. I've shown the base case (gotten on the ladder), then showed how to get from one rung to the next. That's the whole argument.

Example 2: Sums

Induction is often used to prove things about sums, because sums are naturally recursive: To compute the first $n+1$ terms of a sum, add a term on to the sum of the first n terms. Here's a fact that we already know:

For all positive integers n , $1 + 2 + 3 + \dots + n = \frac{n(n+1)}{2}$.

We already proved this in class without induction, by noticing this is a sum of an arithmetic sequence and then using the flip-and-add trick. But let's prove it with induction too.

The predicate here is $P(n)$ which is the statement that $1 + 2 + 3 + \dots + n = \frac{n(n+1)}{2}$. We are claiming $P(n)$ is true for all positive integers.

The framework:

- **Base case:** The smallest input value here is *not* zero this time because it says "all *positive* integers" and 0 isn't positive. The smallest value this time is 1. So is $P(1)$ true? $P(1)$ is the statement that $1 = \frac{1(1+1)}{2}$. The "sum" that would normally be on the left side of the equals sign is just a single term. On the right, you can see that the fraction is actually 1. So the base case holds.
- **Induction hypothesis:** Assume that for some natural number k , $P(k)$ is true; that is, assume that $1 + 2 + 3 + \dots + k = \frac{k(k+1)}{2}$
- **Proof step:** We want to show that $P(k+1)$ is true, that is -- $1 + 2 + 3 + \dots + (k+1) = \frac{(k+1)((k+1)+1)}{2}$
To do this, we will look at the left side and group together the first k terms; that sum has a formula because of the induction hypothesis, so then we add that to $k+1$ and see if it equals the right side.

Here's a completed proof:

To prove: For all positive integers n , $1 + 2 + 3 + \dots + n = \frac{n(n+1)}{2}$.

Proof: The smallest value of the input is $n = 1$. When that happens, on the left side of the equation we just have 1. On the right side, we have $\frac{1(2)}{2}$ and this equals 1. Therefore the left and right sides equal each other when $n = 1$.

Now assume for some positive integer k that

$$1 + 2 + 3 + \dots + k = \frac{k(k+1)}{2}$$

We want to show:

$$1 + 2 + 3 + \dots + (k+1) = \frac{(k+1)((k+1)+1)}{2}$$

Look just at the left side. The next-to-last term on the left is k . Write it as:

$$1 + 2 + 3 + \dots + k + (k+1)$$

Now group off the first k terms:

$$(1 + 2 + 3 + \dots + k) + (k+1)$$

By the induction hypothesis, the first group equals $\frac{k(k+1)}{2}$, so replace:

$$(1 + 2 + 3 + \dots + k) + (k+1) = \frac{k(k+1)}{2} + (k+1)$$

Now the expression on the right can be simplified:

$$\begin{aligned}
\frac{k(k+1)}{2} + (k+1) &= \frac{k(k+1)}{2} + \frac{2k+2}{2} && \text{(Common denominators)} \\
&= \frac{k^2 + k + 2k + 2}{2} && \text{(Multiply out the first numerator)} \\
&= \frac{k^2 + 3k + 2}{2} && \text{(Add like terms)} \\
&= \frac{(k+1)(k+2)}{2} && \text{(Factor)} \\
&= \frac{(k+1)((k+1)+1)}{2} && \text{(Making it look like what we want)}
\end{aligned}$$

This shows that $1 + 2 + 3 + \dots + (k+1) = \frac{(k+1)((k+1)+1)}{2}$. This is what we wanted to show, so we're done. 😊

Example 3: Fibonacci numbers

Recursively-defined sequences like the Fibonacci numbers almost exclusively need induction to prove anything about them. Remember the Fibonacci are 1, 1, 2, 3, 5, 8, 13, 21, 34, . . . You might notice that:

Every third Fibonacci number is even.

How can we prove this with induction? It helps to first rephrase this a little, so a predicate is involved:

Let F_n ($n \geq 1$) be the n th Fibonacci number. (So $F_1 = 1, F_2 = 1, F_3 = 2$, etc.) Then for all positive integers n , F_{3n} is even.

Notice the predicate, $P(n)$ here is now " F_{3n} is even". The expression F_{3n} is an economical way of saying "every third Fibonacci number".

The framework:

- **Base case:** The smallest value here is $n = 1$ ("positive integers"). Is $P(1)$ true? Yes, because $F_{3(1)}$ is F_3 and that's equal to 2 which is even.
- **Induction hypothesis:** Assume that for some positive integer k that $P(k)$ is true. That is, assume F_{3k} is even.
- **Proof step:** We want to show that $P(k+1)$ is true, that is, $F_{3(k+1)}$ is even. A possible strategy is to use the recurrence relation that defines the Fibonacci numbers to rewrite $F_{3(k+1)}$.

That's a good framework. Here's a completed proof:

To prove: Let F_n ($n \geq 1$) be the n th Fibonacci number. (So $F_1 = 1, F_2 = 1, F_3 = 2$, etc.) Then for all positive integers n , F_{3n} is even.

Proof: If $n = 1$, then $F_{3(1)} = F_3 = 2$ which is clearly even. So the base case is satisfied.

Now assume that for some positive integer k , that F_{3k} is even. We want to show that $F_{3(k+1)}$ is even. Expanding out the index gives F_{3k+3} . Use the recurrence relation that defines the Fibonacci numbers to write:

$$F_{3k+3} = F_{3k+2} + F_{3k+1}$$

(That is, F_{3k+3} is the sum of the two previous terms.) Use the recurrence relation again on F_{3k+2} to get:

$$F_{3k+3} = (F_{3k+1} + F_{3k}) + F_{3k+1}$$

Now add like terms:

$$F_{3k+3} = 2F_{3k+1} + F_{3k}$$

The first term on the right is even because it has a factor of 2. The second term, F_{3k} , is even because we assumed that it was even, in the induction hypothesis. Since the sum of two even numbers is another even number, this means that F_{3k+3} is even. That's what we wanted to show.

Therefore every third Fibonacci number is even. 😊

Reality check

You might be feeling a little disoriented right now, because this is the first really formal framework for building general mathematical explanations you've been exposed to. That's OK. This takes getting used to, like most things.

The key things to keep in mind at this point are:

- Mathematical induction is a *framework* for building explanations — that's good! It means that if you can master this framework, there'll be no guesswork as to whether your explanation is satisfactory or not.
- The important thing is to master the *framework* so you can understand the structure of an induction proof — so you can follow the gist of an argument done in this way.
- *Actually writing completed proofs* like the stuff in green above is a secondary issue for us. Mathematicians, and computer scientists interested in the theory of computing, should totally learn to do this and do it well since induction is a fundamental tool for thinking. But, what's important for *all* of us right now is being able to **read and understand a completed proof**. You will more likely need to do this — because you're reading an academic paper, or trying to understand someone else's argument about an algorithm — than you will be constructing completed proofs on your own.
- Reading and understanding a written proof requires careful thought, and **you have to scrutinize every step and make sure you understand**, or else the whole proof won't make sense. So when reading a proof, think of it (or even rewrite it) so each step is like a bullet point in a bad PowerPoint presentation. **Do not move on to the next bullet point until you are completely satisfied you understand the current one, enough so that you could easily explain it to others.**