

21 - Simple Induction

Lean: First Steps

Tariq Rashid

January 17, 2025

Simple Induction

- Induction is a powerful idea that can dramatically simplify some proofs that would otherwise be quite hairy.
- Here we'll see how simple induction can unlock a not-so-simple task.

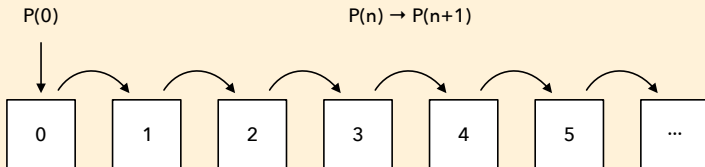
- Prove the following is true for every natural number n ,

$$2^n \geq n + 1$$

- It is not immediately obvious how to do this in our head.

The Idea of Induction

- To prove a proposition P is true for any natural number n , we need to show two things:
 - **Base case:** P is true for $n = 0$.
 - **Inductive step:** P true for n implies P true for $(n + 1)$.



- Let $P(n)$ be the proposition $2^n \geq n + 1$.
- The **base case** $P(0)$ is $2^0 \geq 0 + 1$. Because $2^0 = 1 \geq 1$, the base case $P(0)$ is true.
- The **inductive step** is $P(n) \implies P(n + 1)$. To prove this we assume the **induction hypothesis** $P(n)$ is true and derive $P(n + 1)$.
 - $2^{n+1} = 2 \cdot 2^n$. By assumption $2^n \geq n + 1$, so $2 \cdot 2^n \geq 2 \cdot (n + 1)$.
 - $2 \cdot (n + 1) = (n + 1) + n + 1$ lets us see $(n + 1) + n + 1 \geq (n + 1) + 1$.
 - Putting all this together, we have $2^{n+1} \geq (n + 1) + 1$.
- We have shown $P(0)$ and $P(n) \implies P(n + 1)$, so by induction, $P(n)$ is true for all natural numbers n .

$$P(n) : 2^n \geq n + 1$$

proof objective

base case $P(0)$

$$\begin{aligned} 2^0 &= 1 \\ &\geq 0 + 1 \end{aligned}$$

so $P(0)$ is true

inductive step $P(n) \implies P(n + 1)$

$$P(n) : 2^n \geq n + 1$$

induction hypothesis (1)

$$\begin{aligned} 2^{n+1} &= 2 \cdot 2^n \\ &\geq 2 \cdot (n + 1) \\ &= (n + 1) + 1 + n \\ &\geq (n + 1) + 1 \end{aligned}$$

by (1)

so $P(n + 1)$ is true

$$P(n) : 2^n \geq n + 1$$

by induction

-- 21 - Simple Induction

`import Mathlib.Tactic`

`example {n : ℕ} : 2^n ≥ n + 1 := by`
 `induction n with`
 `| zero =>`
 `norm_num`
 `| succ n ih =>`
 `calc`
 `2^(n + 1) = 2 * 2^n := by ring`
 `_ ≥ 2 * (n + 1) := by rel [ih]`
 `_ = (n + 1) + 1 + n := by ring`
 `_ ≥ (n + 1) + 1 := by norm_num`

- The induction is started using `induction n with` which tells Lean we want to do induction on the variable `n`.
- After that we have two sections, starting with `| zero =>` for the base case, and `| succ n ih =>` for the inductive step.
 - The code indented under `| zero =>` is the proof of the base case $2 \wedge 0 \geq 0 + 1$, which here can be resolved by **norm_num**.
 - The code indented under `| succ n ih =>` is the proof of the inductive step $2 \wedge (n + 1) \geq n + 1 + 1$, which here requires a **calc** section. The inductive hypothesis is established as `ih`.

- Placing the cursor before `induction n with` shows the original proof goal.

$$\vdash 2^n \geq n + 1$$

- Moving the cursor to the start of the next line `| zero =>` shows the goal for the base case.

$$\vdash 2^0 \geq 0 + 1$$

- Placing the curser just before `| succ n ih =>` shows the goal for the inductive step, and also the inductive hypothesis `ih`.

```
ih : 2 ^ n ≥ n + 1
⊢ 2 ^ (n + 1) ≥ n + 1 + 1
```

Easy Exercise

- Write a Lean program to prove, by induction, that

$$3^n \geq n + 1$$

for any natural number n .