Induction, Episode VI: Return of the I.H.

Part a: Recursive Definition

Ian Ludden

Learning Objectives

By the end of this lesson, you will be able to:

Learning Objectives

By the end of this lesson, you will be able to:

- Understand how to read a recursive definition, e.g., compute selected values or objects produced by that definition.

Learning Objectives

By the end of this lesson, you will be able to:

- Understand how to read a recursive definition, e.g., compute selected values or objects produced by that definition.
- Know that a recursive definition, and an inductive proof, require both a base case and an inductive step/formula.

Learning Objectives

By the end of this lesson, you will be able to:

- Understand how to read a recursive definition, e.g., compute selected values or objects produced by that definition.
- Know that a recursive definition, and an inductive proof, require both a base case and an inductive step/formula.
- Define the Fibonacci numbers.

Recursive Definition

Recursive Definition

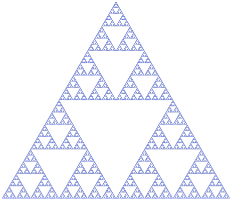
- Recursion: divide problem into smaller problem(s)

Recursive Definition

- Recursion: divide problem into smaller problem(s)
- Coding (mergesort and other "divide-and-conquer" algorithms)

Recursive Definition

- Recursion: divide problem into smaller problem(s)
- Coding (mergesort and other "divide-and-conquer" algorithms)
- Art (esp. fractal art)

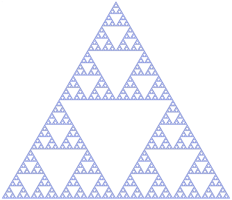


Sierpinski triangle

Droste effect

Recursive Definition

- Recursion: divide problem into smaller problem(s)
- Coding (mergesort and other "divide-and-conquer" algorithms)
- Art (esp. fractal art)
- Closely related to induction



Sierpinski triangle

Droste effect

Some Classic Examples

Example 1: Factorial

base case — recursive formula

Define $f : \mathbb{N} \to \mathbb{N}$ by $f(0) = 1$ and $f(n) = n \cdot f(n-1)$ for $n > 0$.

$f(5) = 5 \cdot f(4) = 5 \cdot 4 \cdot f(3) = \ldots = 5!$ $\quad 0! = 1$

$$f(n) = \begin{cases} 1 & \text{if } n = 0 \\ n \cdot f(n-1) & \text{otherwise } (n > 0) \end{cases}$$

Some Classic Examples

Example 1: Factorial

Define $f : \mathbb{N} \to \mathbb{N}$ by $f(0) = 1$ and $f(n) = n \cdot f(n-1)$ for $n > 0$.

Example 2: Fibonacci

two base cases

Define $F : \mathbb{N} \to \mathbb{N}$ by $F(0) = 0$, $F(1) = 1$, and $F(n) = F(n-1) + F(n-2)$ for $n > 1$.

F_0

0, 1, 1, 2, 3, 5, 8, 13, 21, 34, ...

Preview: Connection to Induction

Preview: Connection to Induction

Example 3: Implicit Summation

Define $g : \mathbb{Z}^+ \to \mathbb{Z}^+$ by

$$g(n) = \begin{cases} 2 & \text{if } n = 1 \\ n(n+1) + g(n-1) & \text{otherwise.} \end{cases}$$

$$g(n) = n(n+1) + (n-1)(n-1+1) + g(n-2)$$

$$= n(n+1) + (n-1)n + (n-2)(n-1) + g(n-3)$$

$$\vdots$$

$$= \sum_{i=1}^{n} i(i+1) = \frac{n(n+1)(n+2)}{3}.$$

"closed form"

Recap: Learning Objectives

By the end of this lesson, you will be able to:

- Understand how to read a recursive definition, e.g., compute selected values or objects produced by that definition.
- Know that a recursive definition, and an inductive proof, require both a base case and an inductive step/formula.
- Define the Fibonacci numbers.