CSCI 136:
Data Structures
and
Advanced Programming

Lecture 12

Mathematical Induction

Instructor: Dan Barowy

Williams

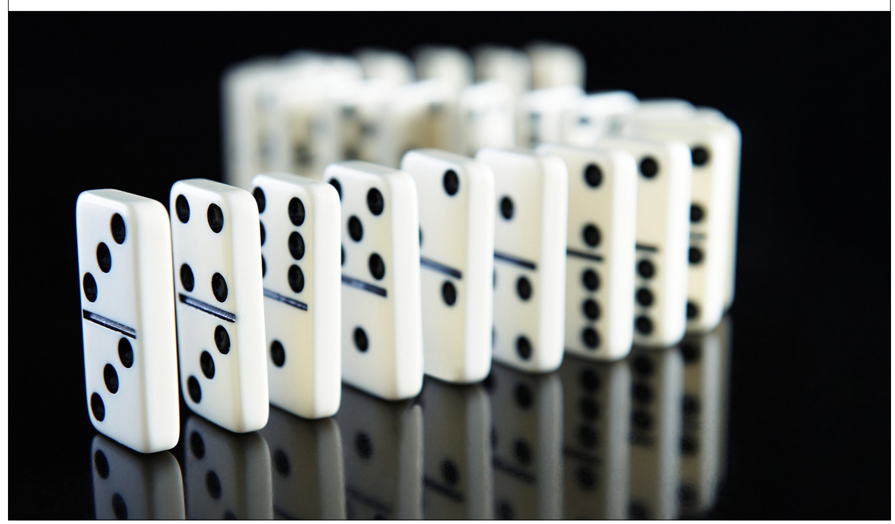
Announcements

- Come to colloquium today— it's never too early to start getting credit! (also, it's fun)
- 2:30pm in Wege Auditorium (TCL 123) every Friday

Outline

1. Mathematical induction
2. Example proofs

Mathematical Induction



A note about "formal methods"

If the problem "fits" the mold, there is a procedure for determining truth.

Mathematical Induction

- The **mathematical cousin** of **recursion** is **induction**
- Induction is a **proof technique**
- Purpose: to **simultaneously prove** an **infinite number** of theorems!

Principle of Mathematical Induction

Let **P(n)** be a **predicate** that is defined for **integers n**, and let **a** be a **fixed integer**.

If the following two statements are **true**:

1. **P(a)** is **true**.
2. For all integers **k ≥ a**, **if P(k)** is **true then P(k + 1)** is **true**.

then the statement

for all integers **n ≥ a**, **P(n)** is **true**

is **also true**.

Principle of Mathematical Induction (variant)

Let **P(n)** be a **predicate** that is defined for **integers n**, and let **a** be a **fixed integer**.

If the following two statements are **true**:

1. **P(a)** is **true**.
2. For all integers **k > a**, **if P(k-1)** is **true then P(k)** is **true**.

then the statement

for all integers **n ≥ a**, **P(n)** is **true**

is **also true**.

To be clear:

If you want to prove that **P(n)** is **true** for all integers **n ≥ a,**

1. You must first prove that **P(a)** is **true**.

2. Then you must prove that:

For all integers **k ≥ a**, **if P(k)** is **true then P(k+1)** is **true**.

Critically, when proving #2, **assume** that **P(k)** is **true** and **show** that **P(k+1) must also be true**.

Names for things and "form"

Hypothesis: **P(n)** is **true** for all integers **n ≥ a,**

1. Base case: **P(a)** is **true**.

2. Inductive step:

For all integers **k ≥ a**, **if P(k)** is **true then P(k+1)** is **true**.

Like recursion, there is an analogy

Like recursion, there is an analogy

Example

Prove that the sum of the first n integers is:

$$\frac{n(n+1)}{2}$$

Example

Put another way, prove

$$\mathbf{P(n)} : 1 + 2 + 3 + \ldots + n = \frac{n(n+1)}{2}$$

for all $n \geq 1$.

We have an unbounded number of hypotheses ("for all $n \geq 1$").

Use mathematical induction.

Remember the template!

Step 1: Prove **P(a)**

Step 2: Prove **P(k) ⇒ P(k+1)**

Therefore,

$$\mathbf{P(n)} : 1 + 2 + 3 + \ldots + n = \frac{n(n+1)}{2}$$

For all $n \geq 1$.

Is **true**.

Example

Step 1: Prove **P(a)**

What would a good **a** be?

$$P(n) : 1 + 2 + 3 + \ldots + n = \frac{n(n+1)}{2}$$

The "simplest" instance is **a = 1**. Let's start there.

Example

Step 1: Prove **P(a)**

$$P(a) : 1 = \frac{1(1+1)}{2}$$

Is this statement true? **Yes.**

Proof: $\frac{1(1+1)}{2} = \frac{2}{2} = 1$

Example

Step 2: Prove **P(k) ⇒ P(k+1)**

Assume the following is true:

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

Prove:

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

Example

Step 2: Prove **P(k) ⇒ P(k+1)**

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

Let's handle the left side first.

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

Looks familiar. Isn't it the same as:

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

Example

Step 2: Prove **P(k) ⇒ P(k+1)**

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

According to P(k), which is true,
it must be equal to:

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

Example

Step 2: Prove **P(k) ⇒ P(k+1)**

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

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

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

Let's stop here.
The left side is
$$= \frac{(k+1)(k+2)}{2}$$

Example

Step 2: Prove **P(k) ⇒ P(k+1)**

P(k+1) : $1 + 2 + 3 + ... + (k + 1) = \frac{(k+1)((k+1)+1)}{2}$

Let's handle the right side now.

$$\frac{(k+1)((k+1)+1)}{2}$$

Simplify

$$\frac{(k+1)(k+2)}{2}$$

Let's stop here.

Example

Step 2: Prove **P(k) ⇒ P(k+1)**

P(k+1) : $1 + 2 + 3 + ... + (k + 1) = \frac{(k+1)((k+1)+1)}{2}$

We just showed that the left side

$$\frac{(k+1)(k+2)}{2}$$

equals the right side

$$\frac{(k+1)(k+2)}{2}$$

Example

Step 1: Prove **P(a)** ✓

Step 2: Prove **P(k) ⇒ P(k+1)** ✓

Therefore,

P(n) : $1 + 2 + 3 + ... + n = \frac{n(n+1)}{2}$

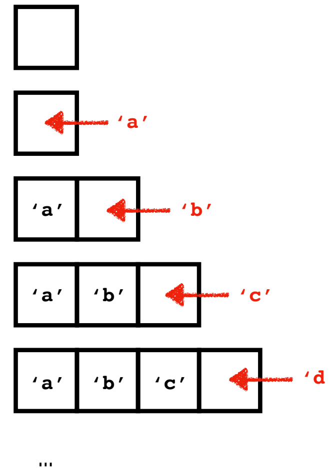
For all $n \geq 1$.

Is **true**. ✓

Expanding vectors: why double?

Why is the **array doubling** strategy for Vector **better** than expanding the array **one element at a time**?

One-at-a-time expansion



Initial array.

Insert element.

New array; **copy** previous; **insert** element.

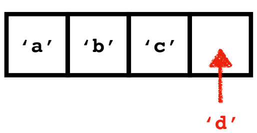
New array; **copy** previous; **insert** element.

New array; **copy** previous; **insert** element.

...

Insertion into an array

How much does **array insertion** cost?



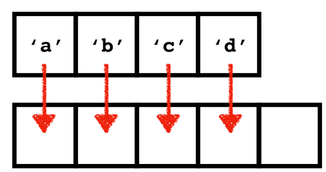
It costs **O(1)**.

In fact, lookup and insertion both cost **O(1)**.

Tradeoff: arrays are fixed size.

Copying an array

How much does an **array copy** cost?

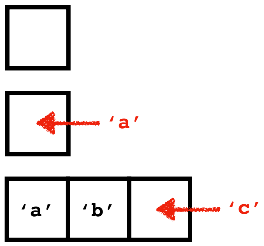


It costs **O(1)** × **m**, where **m** is the size of the original array.

≈ **O(m)**

One-at-a-time expansion costs?

(in the worst case, each time)



Initial array.

Insert element.

New array; **copy** previous; **insert** element.

O(m) + **O(1)** ≈ **O(m)**, where **m** is the size of the original array.

Cost is **dominated by the size of the array** being copied.

How many copies?

of copies for one-at-a-time expansion:

	1	**+**	**2**	**+**	**3**	**+**	**…**	**+(n-1)**
`add()`	2nd elem.		3rd elem.		4th elem.			nth elem.

Recall theorem: $1 + 2 + 3 + \ldots + k = k(k+1)/2$

Sub n-1 for k: $(n-1)((n-1)+1)/2 = n(n-1)/2$

$= n^2/2 - n/2$

One-at-a-time expansion costs ≈ **O(n²)**

How many copies?

of copies for doubling expansion:

	1	**+**	**2**	**+**	**4**	**+**	**…**	**+(n/2)**
`add()`	up to 2nd elem.		up to 4th elem.		up to 8th elem.			up to nth elem.

Neat theorem: $1 + 2 + 4 + \ldots + 2^{k-1} = 2^k - 1$

Suppose $n = 2^k$.

Then $1 + \ldots + n/2 = 1 + \ldots + 2^k/2$

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

Doubling expansion costs ≈ **O(n)**

Which is faster?

💩 One-at-a-time expansion costs ≈ **O(n²)** 💩

😎 Doubling expansion costs ≈ **O(n)** 😎

Doubling is Vin Diesel-approved.

Activity

Prove: n cents can be obtained by using only 3-cent and 8-cent coins, for all $n \geq 15$.

Recap & Next Class

Today we learned:

Mathematical induction

Next class:

Sorting