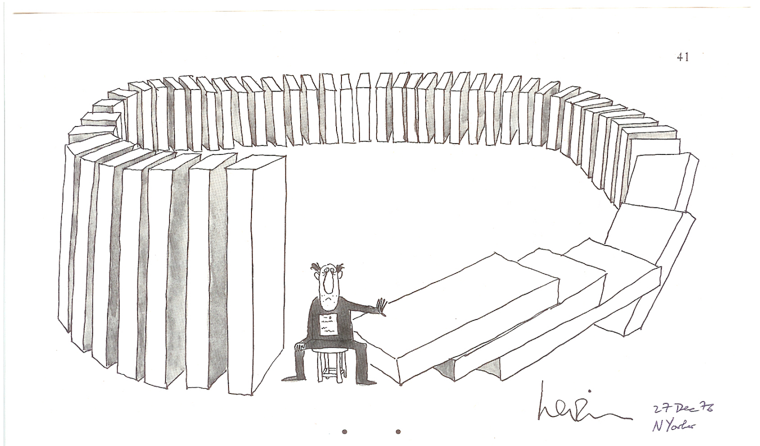


CSCI 2824: Discrete Structures

Lecture 22: Matrices. Complexity. Weak Induction

Reminders

Submissions:

- Homework 8: **Fri 10/25 at noon –** Gradescope
- Quizlet : **Fri 10/18 at 8pm**

Readings:

- Ch. 3 – Algorithms
    - 3.3 Complexity of Algorithms – Matrices
- Ch. 5 – Induction and Recursion
    - 5.1 Mathematical Induction

Last time

Algorithms…

- Big O, Big Omega, Big Theta
- Matrices
  - Addition

Today:

- Finishing Matrices
- Starting Induction

Deductive and induction reasoning

Logic: another thing that penguins aren't very good at.

Matrices and matrix operations

$$\begin{aligned} 3x + 4y + 5z &= 1 \\ 2x + 8y + 3z &= 2 \\ 4x + 2y + 2z &= 3 \end{aligned} \quad \Leftrightarrow \quad \begin{bmatrix} 3 & 4 & 5 \\ 2 & 8 & 3 \\ 4 & 2 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

- Rectangular thing is a **matrix**, tall skinny things are **vectors**
- **Definition:** A matrix with m rows and n columns has **dimensions** $m \times n$
- **Definition:** A vector with n entries has **length** n
- **Notation:** Matrices are represented by capital letters, like A and M.
  Vectors are represented by lowercase letters like $\mathbf{x}$ and $\mathbf{b}$
  (often bold-faced)
- **Example:** The above **matrix equation** could be written as $A\mathbf{x} = \mathbf{b}$

Matrices and matrix operations

- Matrices and vectors can be **added** and **multiplied** (but not divided)
- **Definition:** The **sum** of matrices *A* and *B* is the matrix obtained by adding the corresponding entries of each matrix together
- **Example:**

$$\begin{bmatrix} 3 & 4 & 5 \\ 2 & 8 & 3 \\ 4 & 2 & 2 \end{bmatrix} + \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} = \begin{bmatrix} 4 & 6 & 8 \\ 6 & 13 & 9 \\ 11 & 10 & 11 \end{bmatrix}$$

- **Note:** Only makes sense if *A* and *B* have the **same** dimensions!
- **Notation:** We refer to the entry in the i^{th} row and j^{th} column of the matrix *A* as a_{ij}, or $A[i, j]$.

Matrices and matrix operations

Complexity of matrix addition:

- Straightforward: We add each pair of entries.

  ⇒ for two $m \times n$ matrices, there are $m*n$ entries, so $m*n$ additions

  ⇒ for square matrices of size $n \times n$, that's n^2 additions, so this is $O(n^2)$

Matrices and matrix operations

Complexity of matrix addition:

- Using the slick pseudocode method (for adding two square matrices):

```
def matrixAdd (A, B):
    S = "0"                    # initialize as n × n zero matrix
    for i in 1,num_rows:
        for j in 1,num_cols:
            S[i,j] = A[i,j] + B[i,j]
    return (S)
```

Turn loops into sums and count up the basic operations:

Complexity:

$$\sum_{i=1}^{n}\sum_{j=1}^{n} 1 = \sum_{i=1}^{n} n = n^2$$

Matrices and matrix operations

- Matrices can also **multiply** vectors, resulting in a new vector.

$$\begin{aligned} 3x + 4y + 5z &= 1 \\ 2x + 8y + 3z &= 2 \\ 4x + 2y + 2z &= 3 \end{aligned} \Leftrightarrow \begin{bmatrix} 3 & 4 & 5 \\ 2 & 8 & 3 \\ 4 & 2 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

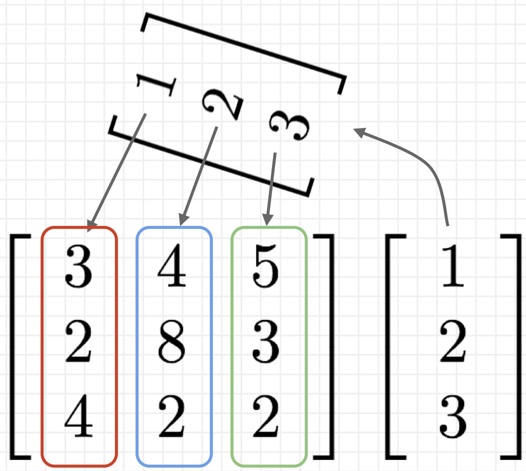
- This operation is defined directly out of the analogy between matrix equations and linear systems of equations.

$$\begin{matrix} 3x + 4y + 5z \\ 2x + 8y + 3z \\ 4x + 2y + 2z \end{matrix} \Leftrightarrow \begin{bmatrix} 3 & 4 & 5 \\ 2 & 8 & 3 \\ 4 & 2 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

Matrices and matrix operations

- Think of it as taking the vector and setting it down on top of the matrix.

$$\begin{bmatrix} 3 & 4 & 5 \\ 2 & 8 & 3 \\ 4 & 2 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} = \begin{bmatrix} 3 \cdot 1 + 4 \cdot 2 + 5 \cdot 3 \\ 2 \cdot 1 + 8 \cdot 2 + 3 \cdot 3 \\ 4 \cdot 1 + 2 \cdot 2 + 2 \cdot 3 \end{bmatrix} = \begin{bmatrix} 26 \\ 27 \\ 14 \end{bmatrix}$$



Important rule: This means that the length of the vector **must equal** the number of columns of the matrix.

Matrices and matrix operations

Pseudocode and complexity: intuition and estimation – counting multiplications/additions, what is a rough estimate of the complexity?

$$\begin{bmatrix} 1 & 3 \\ -2 & 4 \\ 1 & 5 \end{bmatrix} \begin{bmatrix} 2 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \cdot 2 + 3 \cdot 1 \\ -2 \cdot 2 + 4 \cdot 1 \\ 1 \cdot 2 + 5 \cdot 1 \end{bmatrix} = \begin{bmatrix} 2 + 3 \\ -4 + 4 \\ 2 + 5 \end{bmatrix} = \begin{bmatrix} 5 \\ 0 \\ 7 \end{bmatrix}$$

Matrices and matrix operations

Pseudocode and complexity: intuition and estimation – counting multiplications/additions, what is a rough estimate of the complexity?

```
In [244]: y = []               # initialize output vector
     ...: n = len(A)
     ...: for i in range(0,n):
     ...:   y.append(A[i][0]*x[0])
     ...:   for j in range(1,n):
     ...:       y[i] = y[i] + A[i][j]*x[j]
```

$$\begin{bmatrix} 1 & 3 \\ -2 & 4 \\ 1 & 5 \end{bmatrix} \begin{bmatrix} 2 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \cdot 2 + 3 \cdot 1 \\ -2 \cdot 2 + 4 \cdot 1 \\ 1 \cdot 2 + 5 \cdot 1 \end{bmatrix} = \begin{bmatrix} 2 + 3 \\ -4 + 4 \\ 2 + 5 \end{bmatrix} = \begin{bmatrix} 5 \\ 0 \\ 7 \end{bmatrix}$$

Matrices and matrix operations

Pseudocode and complexity: intuition and estimation – counting multiplications/additions, what is a rough estimate of the complexity?

```
In [244]: y = []              # initialize output vector
     ...: n = len(A)
     ...: for i in range(0,n):
     ...:   y.append(A[i][0]*x[0])
     ...:   for j in range(1,n):
     ...:       y[i] = y[i] + A[i][j]*x[j]
```

- Count additions and multiplications. (Usually, we count FLOPs (floating point operations) instead.)
- **Complexity:** $$= \sum_{i=1}^{n} 1 \sum_{j=1}^{n} 2 = \sum_{i=1}^{n} 2n = 2n^2$$

Matrix Multiplication

Multiplying matrices: think of it as doing a few matrix-vector multiplications

$$\begin{bmatrix} 1 & 3 & -1 \\ -2 & 4 & 0 \\ 1 & 5 & 2 \end{bmatrix} \begin{bmatrix} \boxed{\begin{matrix} 2 \\ 3 \\ 1 \end{matrix}} & \boxed{\begin{matrix} 1 \\ -1 \\ 0 \end{matrix}} & \boxed{\begin{matrix} 0 \\ 2 \\ -2 \end{matrix}} \end{bmatrix} = ?$$

$$AB = A[\mathbf{b}_1 \ \ \mathbf{b}_2 \ \ \mathbf{b}_3] = [A\mathbf{b}_1 \ \ A\mathbf{b}_2 \ \ A\mathbf{b}_3]$$

$$\begin{bmatrix} 1 & 3 & -1 \\ -2 & 4 & 0 \\ 1 & 5 & 2 \end{bmatrix} \begin{bmatrix} 2 & 1 & 0 \\ 3 & -1 & 2 \\ 1 & 0 & -2 \end{bmatrix} = \begin{bmatrix} 1 \cdot 2 + 3 \cdot 3 + -1 \cdot 1 & \_ & \_ \\ \_ & \_ & \_ \\ \_ & \_ & \_ \end{bmatrix}$$

$$\begin{bmatrix} 1 & 3 & -1 \\ -2 & 4 & 0 \\ 1 & 5 & 2 \end{bmatrix} \begin{bmatrix} 2 & 1 & 0 \\ 3 & -1 & 2 \\ 1 & 0 & -2 \end{bmatrix} = \begin{bmatrix} 1 \cdot 2 + 3 \cdot 3 + -1 \cdot 1 & \_ & \_ \\ \_ & \_ & \_ \\ \_ & \_ & \_ \end{bmatrix}$$

$$\begin{bmatrix} 1 & 3 & -1 \\ -2 & 4 & 0 \\ 1 & 5 & 2 \end{bmatrix} \begin{bmatrix} 2 & 1 & 0 \\ 3 & -1 & 2 \\ 1 & 0 & -2 \end{bmatrix} = \begin{bmatrix} 10 & \_ & \_ \\ -2 \cdot 2 + 4 \cdot 3 + 0 \cdot 1 & \_ & \_ \\ \_ & \_ & \_ \end{bmatrix}$$

$$\begin{bmatrix} 1 & 3 & -1 \\ -2 & 4 & 0 \\ 1 & 5 & 2 \end{bmatrix} \begin{bmatrix} 2 & 1 & 0 \\ 3 & -1 & 2 \\ 1 & 0 & -2 \end{bmatrix} = \begin{bmatrix} 1 \cdot 2 + 3 \cdot 3 + -1 \cdot 1 & \_ & \_ \\ \_ & \_ & \_ \\ \_ & \_ & \_ \end{bmatrix}$$

$$\begin{bmatrix} 1 & 3 & -1 \\ -2 & 4 & 0 \\ 1 & 5 & 2 \end{bmatrix} \begin{bmatrix} 2 & 1 & 0 \\ 3 & -1 & 2 \\ 1 & 0 & -2 \end{bmatrix} = \begin{bmatrix} 10 & \_ & \_ \\ -2 \cdot 2 + 4 \cdot 3 + 0 \cdot 1 & \_ & \_ \\ \_ & \_ & \_ \end{bmatrix}$$

$$\begin{bmatrix} 1 & 3 & -1 \\ -2 & 4 & 0 \\ 1 & 5 & 2 \end{bmatrix} \begin{bmatrix} 2 & 1 & 0 \\ 3 & -1 & 2 \\ 1 & 0 & -2 \end{bmatrix} = \begin{bmatrix} 10 & \_ & \_ \\ 8 & \_ & \_ \\ 1 \cdot 2 + 5 \cdot 3 + 2 \cdot 1 & \_ & \_ \end{bmatrix}$$

$$\begin{bmatrix} 1 & 3 & -1 \\ -2 & 4 & 0 \\ 1 & 5 & 2 \end{bmatrix} \begin{bmatrix} 2 & 1 & 0 \\ 3 & -1 & 2 \\ 1 & 0 & -2 \end{bmatrix} = \begin{bmatrix} 10 & -2 & 8 \\ 8 & -6 & 8 \\ 19 & -4 & 6 \end{bmatrix}$$

- **Question:** What must be the dimensions of *A* and *B* for this to work?

$$\begin{bmatrix} 1 & 3 & -1 \\ -2 & 4 & 0 \\ 1 & 5 & 2 \end{bmatrix} \begin{bmatrix} 2 & 1 & 0 \\ 3 & -1 & 2 \\ 1 & 0 & -2 \end{bmatrix} = \begin{bmatrix} 10 & -2 & 8 \\ 8 & -6 & 8 \\ 19 & -4 & 6 \end{bmatrix}$$

- **Question:** What must be the dimensions of *A* and *B* for this to work?

  Answer: Need the # columns of *A* to match the # rows of *B*.

- **Question:** What are the dimensions of $C = AB$?

$$\begin{bmatrix} 1 & 3 & -1 \\ -2 & 4 & 0 \\ 1 & 5 & 2 \end{bmatrix} \begin{bmatrix} 2 & 1 & 0 \\ 3 & -1 & 2 \\ 1 & 0 & -2 \end{bmatrix} = \begin{bmatrix} 10 & -2 & 8 \\ 8 & -6 & 8 \\ 19 & -4 & 6 \end{bmatrix}$$

- **Question:** What must be the dimensions of *A* and *B* for this to work?

  Answer: Need the # columns of *A* to match the # rows of *B*.

- **Question:** What are the dimensions of *C* = *AB* ?

  Answer: *C* = *AB* will have the same # rows as *A* and # columns as *B*.

- **Summary:** If *A* is *n x k* and *B* is *k x m* then *C* = *AB* is *n x m*.

Complexity: Multiply *A* and *B*, both *n x n* matrices.

$$AB = A[\mathbf{b}_1 \ \mathbf{b}_2 \ \mathbf{b}_3] = [A\mathbf{b}_1 \ A\mathbf{b}_2 \ A\mathbf{b}_3]$$

Complexity: Multiply *A* and *B*, both *n x n* matrices.

$$AB = A[\mathbf{b}_1 \ \mathbf{b}_2 \ \mathbf{b}_3] = [A\mathbf{b}_1 \ A\mathbf{b}_2 \ A\mathbf{b}_3]$$

- Each column of $C = AB$ is a "mat-vec"
- We saw these each require ~$2n^2$ FLOPs
- And we have n columns to do

⇒ Total mat-mat is ~$n \times 2n^3 = 2n^3$ FLOPs

Summary:

- Matrix addition is $O(n^2)$
- Matrix-vector multiplication (mat-vec) is $O(n^2)$
- Matrix-matrix multiplication (mat-mat) is $O(n^3)$

Algorithm complexity and matrix operations

Extra special FYOG: There is a special kind of a square matrix called **<u>lower-triangular</u>**, where all of the elements above the **main diagonal** are 0.

- **Example:**

$$L = \begin{bmatrix} 1 & 0 & 0 \\ -2 & 4 & 0 \\ 1 & 5 & 2 \end{bmatrix}$$

S'pose we want to compute $L\mathbf{x}$, where $\mathbf{x}$ is an appropriately-sized vector. Any time an entry of $\mathbf{x}$ hits one of the upper 0s, we already know the result will be 0. So we don't want to waste time computing those multiplications.

- **TODO:** Modify your mat-vec code to skip the unnecessary multiplications, assuming a lower-triangular matrix is used.
- **TODO:** Determine the complexity of the new algorithm when L is n x n.

Induction

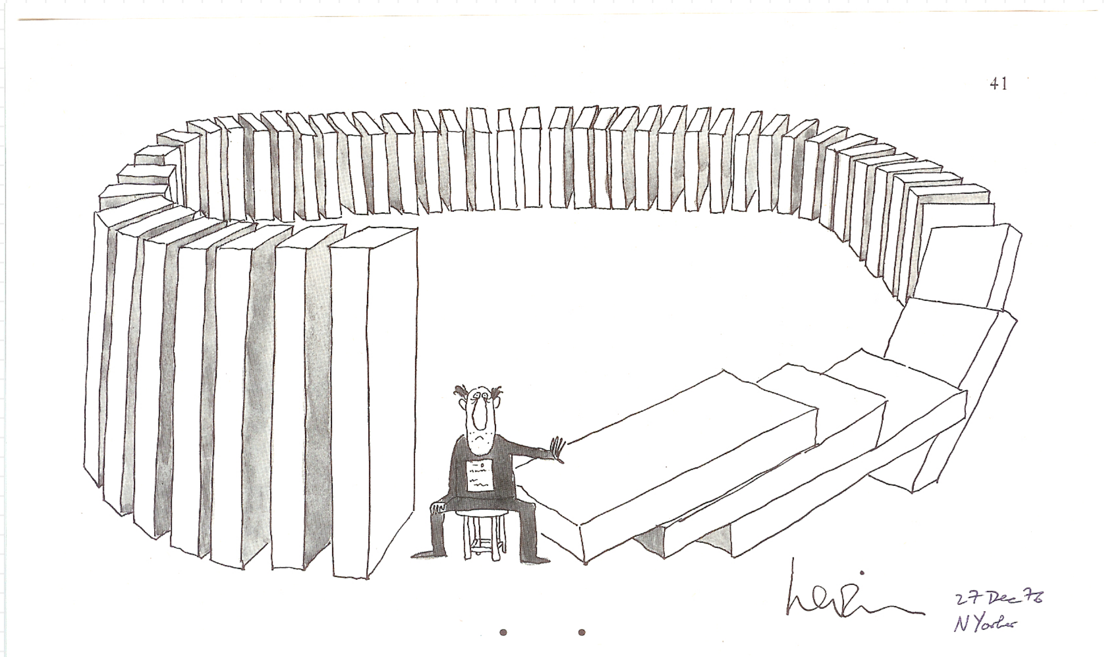
Mathematical induction

- Mathematical induction is a common and powerful way to prove properties of
  - Natural numbers
  - Sets
  - Relations
  - Trees and graphs (we'll get to these in a few weeks!)

- Let's start with a motivating example though, with (almost) no math!

Mathematical induction

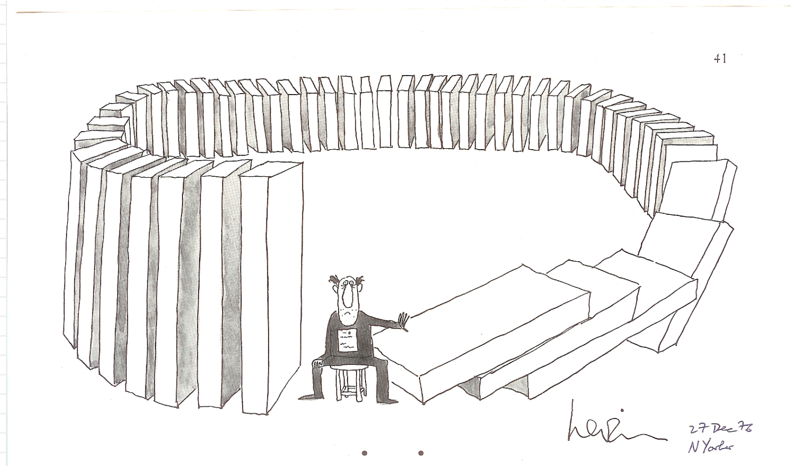
S'pose you have an infinite line of dominos. (set up "correctly", i.e., not too far apart)



Prove that if you tip over the first domino, then the rest of them will fall.

Mathematical induction

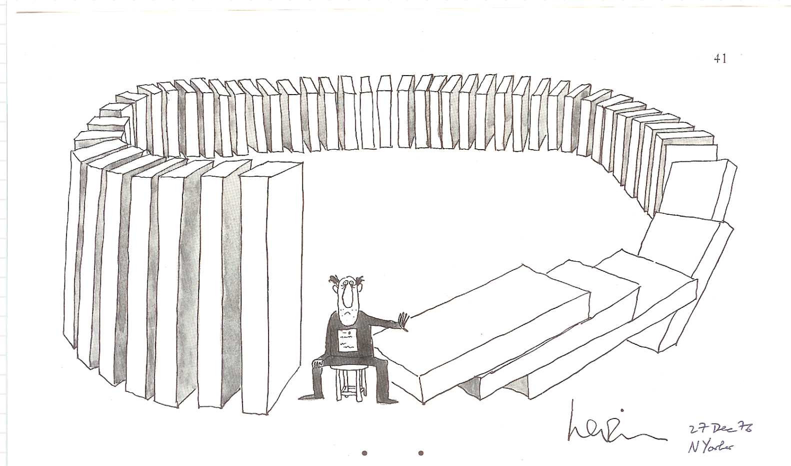
Your argument could go something like this:



First: The first domino falls (because you knocked it over)
Then: Whenever domino number *k* falls, the one after it numbered *k*+1 also will fall (because domino *k* knocks it over).
Therefore: We conclude that all of the dominoes will fall over.

Mathematical induction

Your argument could go something like this:



Base case: The first domino falls (because you knocked it over)

Inductive Step: Whenever domino number *k* falls, the one after it numbered *k*+1 also will fall (because domino *k* knocks it over).

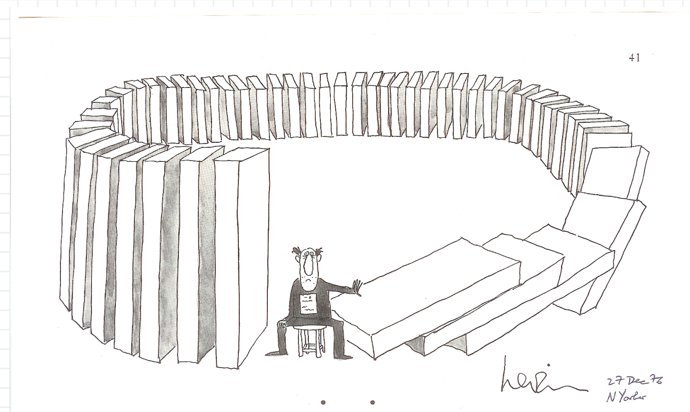
Therefore: We conclude that all of the dominoes will fall over.

Mathematical induction

That kind of argument is the crux of **induction**. To prove a property holds for all natural numbers *k*, we argue as follows:

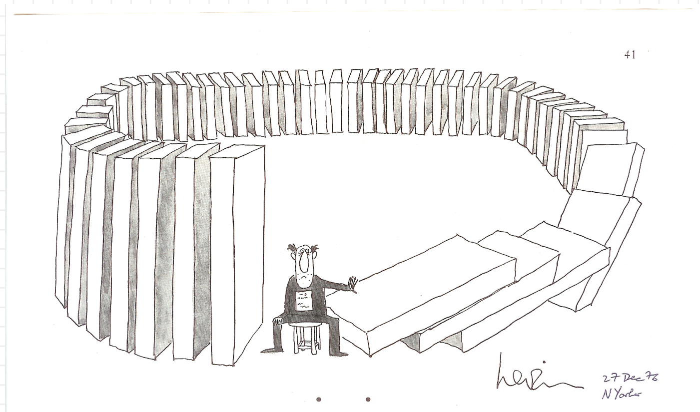
1. The property is true for $k = 0$ (or $k = 1$, or some other **base case**)
2. If the property is true for some natural number *k*, then it is true for natural number $k + 1$

(note that this is asking us to prove a conditional: [property true for *k*] → [true for *k*+1])



Mathematical induction

Example: Prove that $1 + 2 + \cdots + n = \dfrac{n(n+1)}{2}$



Mathematical induction

Example: Prove that $1 + 2 + \cdots + n = \frac{n(n+1)}{2}$

Base case:

Induction step:

Mathematical induction

Example: Prove that $1 + 2 + \cdots + n = \dfrac{n(n+1)}{2}$

Base case: This formula holds for n=1: 1 = (1)(1+1)/2 = 1

Induction step: This requires a proof of the conditional

[formula works for $n=k$] → [formula works for $n=k+1$]

In general, the hypothesis of this conditional, "property holds for $n=k$", is known as the **inductive hypothesis**

So, s'pose the formula works for $n = k$ (**inductive hypothesis**)

Mathematical induction

Example: Prove that $1 + 2 + \cdots + n = \dfrac{n(n+1)}{2}$

Base case: This formula holds for n=1: 1 = (1)(1+1)/2 = 1

Induction step: This requires a proof of the conditional

[formula works for $n=k$] → [formula works for $n=k+1$]

So, s'pose the formula works for $n = k$ (**inductive hypothesis**)

Mathematical induction

Example: Prove that $1 + 2 + \cdots + n = \dfrac{n(n+1)}{2}$

Base case: This formula holds for n=1: 1 = (1)(1+1)/2 = 1

Induction step: This requires a proof of the conditional

[formula works for $n=k$] → [formula works for $n=k+1$]

So, s'pose the formula works for $n = k$ (**inductive hypothesis**)

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

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

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

Mathematical induction

So if we let *P(n)* be the property that we're trying to prove, where *n* is some natural number, then an inductive argument goes like this:

1. **Base case:** Verify that *P(0)* holds (or *P(1)*, or *P(whatever)*)
2. **Induction step:** $(\forall k \geq 0)$ **if** $P(k)$ **then** $P(k+1)$
3. **Conclusion:** $(\forall n \geq 0)$ $P(n)$

There are two slightly different kinds of inductive arguments...

Mathematical induction

There are two slightly different kinds of inductive arguments...

If the argument is of the form

- Verify that $P(1)$ is true
- Assume $P(k)$ is true, and show that $P(k+1)$ must be true (the proof part)

then we call it **weak induction**, or **ordinary induction**.

If the argument is of the form

- Verify that $P(1)$ is true
- Assume $P(k)$ is true for all $k = 1, 2, \ldots, n$, and show $P(n+1)$

then we call it **strong induction**.

Mathematical induction

Example: Propose a formula for the sum of the first *n* odd positive integers. Then prove it using induction.

Exploration:

Conjecture:

Mathematical induction

Example: Propose a formula for the sum of the first *n* odd positive integers. Then prove it using induction.

Exploration:

$$1 = 1$$

$$1 + 3 = 4$$

$$1 + 3 + 5 = 9$$

Conjecture: $1 + 3 + 5 + \ldots + (2n - 1) = n^2$

Now it's show time!

Mathematical induction

Example: Propose a formula for the sum of the first n odd positive integers. Then prove it using induction.

To show: $1 + 3 + 5 + \ldots + (2n - 1) = n^2$

Proof: (by weak induction)

Base case:

Inductive step:

Mathematical induction

Example: Propose a formula for the sum of the first n odd positive integers. Then prove it using induction.

To show: $1 + 3 + 5 + \ldots + (2n - 1) = n^2$

Proof: (by weak induction)

Base case: $n=1$, $1 = 1^2$

Inductive step: S'pose the formula holds for $n=k$ (that is, $1 + 3 + 5 + \ldots + (2k - 1) = k^2$)

$$\Rightarrow 1 + 3 + 5 + \ldots + (2k - 1) + (2k + 1) = k^2 + (2k + 1)$$

$$\Rightarrow 1 + 3 + 5 + \ldots + (2k - 1) + (2k + 1) = (k + 1)^2$$

Conclusion: We've shown that the formula holds for $n=k+1$, thus, by induction, we've proved that the formula is true in general. ⧅

Mathematical induction

Example: Prove that if n is an integer and $n \geq 4$, then $2^n < n!$

Base case:

Inductive step:

Conclusion:

Mathematical induction

Example: Prove that if n is an integer and $n \geq 4$, then $2^n < n!$

Base case: $n=4$, we have $2^{(4)} = 16$ and $4! = 24$, so it is true that $2^4 < 4!$

Inductive step: S'pose the formula holds for $n=k$ $(2^k < k!)$

$$\Rightarrow \; 2^k < k!$$

$$\Rightarrow \; 2^k \cdot 2 < k! \cdot 2$$

$$\Rightarrow \; 2^{k+1} < k! \cdot 2 < k! \cdot (k+1) \qquad (k+1 \text{ definitely} > 2, \text{ since } k \geq n \geq 4)$$

$$\Rightarrow \; 2^{k+1} < (k+1)!$$

Conclusion: We've shown that the inequality holds for $n=k+1$, thus, by induction, we've proved the inequality. ⧅

Mathematical induction

Example: (geometric progressions) Prove that when $r \neq 1$,

$$a + ar + ar^2 + \cdots + ar^n = \frac{ar^{n+1} - a}{r - 1}$$

Base case:

Inductive step:

Mathematical induction

Example: (geometric progressions) Prove that when $r \neq 1$,

$$a + ar + ar^2 + \cdots + ar^n = \frac{ar^{n+1} - a}{r - 1}$$

Base case: $n=0$, $$a = \frac{ar^1 - a}{r - 1} = \frac{a(r-1)}{r-1} = a$$

Inductive step: S'pose the formula holds for $n=k$. Then…

$$\Rightarrow \quad a + ar + ar^2 + \cdots + ar^k + ar^{k+1} = \frac{ar^{k+1} - a}{r - 1} + ar^{k+1}$$

$$= \frac{ar^{k+1} - a}{r - 1} + \frac{ar^{k+1}(r-1)}{r - 1}$$

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

Conclusion:
Thus, by induction, the formula is true ∎

Mathematical induction

Example: Prove that if $n \geq 1$ is an integer, then $n^3 - n$ is divisible by 3

Mathematical induction

Example: Prove that if $n \geq 1$ is an integer, then $n^3 - n$ is divisible by 3

Base case: $n=1 \Rightarrow n^3-n = 1^3-1 = 0$, and $0=3(0)$ so 1^3-1 is divisible by 3.

Inductive step: S'pose k^3-k is divisible by 3, for some integer $k > 1$ (**inductive hypothesis**)
To show: $(k+1)^3 - (k+1)$ must also be divisible by 3.

$$
\begin{aligned}
\Rightarrow \quad (k+1)^3 - (k+1) &= (k^3 + 3k^2 + 3k + 1) - k - 1 \\
&= k^3 - k + 3k^2 + 3k \\
&= 3(\text{some integer}) + 3k^2 + 3k \qquad \leftarrow \text{by } \textbf{inductive hypothesis} \\
&= 3(\text{some integer} + k^2 + k)
\end{aligned}
$$

Therefore, $(k+1)^3 - (k+1)$ is divisible by 3, and the hypothesis is true by **weak induction** ∎

Mathematical induction

Recap:

- **Math induction** -- the proof technique that's a proof by cases on steroids
  - **Base case:** show that the hypothesis is true for the first case
  - **Inductive step:** name your inductive hypothesis ⇒ S'pose true at k, **To Show** true at k+1

Next time:

- We make our induction ***strong***

smbc-comics.com