

# Induction

Used in:

- Design of combinatorial algorithms
- Proof of theorems

Induction is common to both

Use known mathematical proof methods to help create algorithms

## How Does Induction Work

**T** is a theorem.

- If we show that:

1. **T** holds for  $n = 1$  (or a small number)

2. For every  $n > 1$

If **T** holds for  $n - 1$

then **T** holds for  $n$

- THEN theorem **T** holds for *all*  $n \geq 1$

The assumption that **T** holds for  $n - 1$  is called *induction hypothesis*

### Example 1

**Theorem.** For all natural numbers  $p > 1$  and  $n \geq 1$ ,  $p^n - 1$  is divisible by  $p - 1$ .

*Proof:*

Induction base: for  $n = 1$ ,  $p - 1$  is divisible by  $p^1 - 1 = p - 1$

Induction hypothesis: Assume  $p^n - 1$  is divisible by  $p - 1$ .

We have to show that  $p^{n+1} - 1$  is divisible by  $p - 1$ .

Indeed,  $p^{n+1} - 1 = p(p^n - 1) + (p - 1)$ .

By induction hypothesis,  $p^n - 1$  is divisible by  $p - 1$ .

Obviously,  $(p - 1)$  is divisible by  $p - 1$ .

Consequently, we can conclude that  $p^n - 1$  is divisible by  $p - 1$  for all  $n \geq 1$ , that is, the theorem holds.  $\square$

## Example 2

**Theorem.**  $\sum_{i=1}^n i = n(n+1)/2$ .

*Proof:*

Induction base:  $n = 1$  :

$$1 = \sum_{i=1}^1 i = 1(1+1)/2 = 1.$$

Induction hypothesis: Assume  $\sum_{i=1}^k i = k(k+1)/2$ .

We want to show that  $\sum_{i=1}^{k+1} i = (k+1)(k+2)/2$ .

Indeed,

$$\begin{aligned} \sum_{i=1}^{k+1} i &= \sum_{i=1}^k i + (k+1) = \underbrace{k(k+1)/2}_{\text{induction hypothesis}} + (k+1) \\ &= (k+1)(k+2)/2 = (k+1)((k+1)+1)/2. \end{aligned} \quad \square$$

## Strong Induction

- A statement  $\mathbf{T}$  with a parameter  $n$  is true for  $n = 1$  and
- if, for every  $n > 1$  the following holds:

$\mathbf{T}$  being true for every natural number  $m < n$  implies that  $\mathbf{T}$  is true for  $n$ ,

- then  $\mathbf{T}$  is true for all natural numbers

Also, *power(2) induction*: induction on natural numbers that are integer powers of 2

*Multiple induction*: base case for one parameter required another induction on another parameter.

## Example: A Simple Coloring problem

- In general, it is possible to color any planar map with four colours
- The regions formed by (infinite) lines have special properties

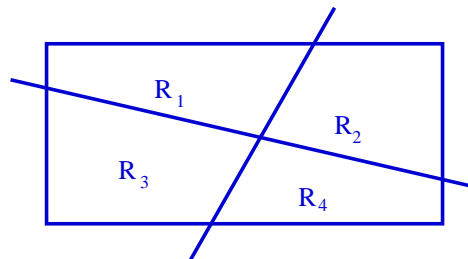
**Theorem.** It is possible to color the regions formed by any number of lines in the plane with only two colours.

\* By *coloring* we mean neighboring regions (regions that have common boundary) have different colors.

*Example:*

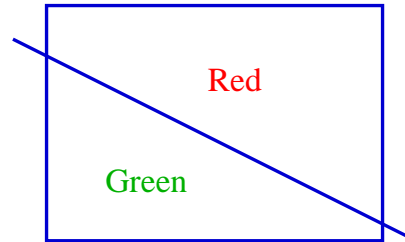
$r_2, r_3$  are neighboring regions of  $r_1$

$r_4$  is not a neighboring region of  $r_1$ .

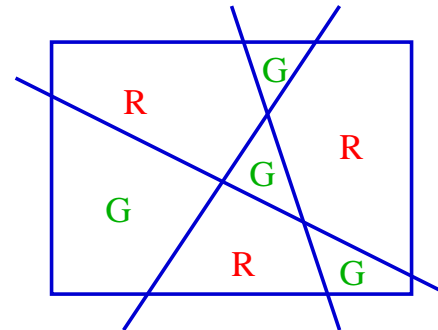
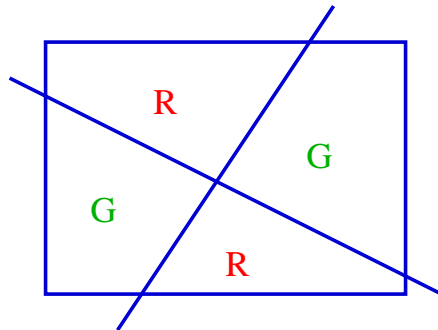


## Proof

- Induction base: We can color regions formed by one line in a plane with 2 color.



- Induction hypothesis: Assume we can color the regions formed by  $< N$  lines in a plane with 2 colors.
- Try ADD A LINE ( $n$  lines now)



## Add a line

- Divide regions into two groups according to which side of the new line they are on
- Leave colors of all regions on one side (say above) the same
- Reverse colors on the other side (below)

Proof (continued):

Consider two neighboring regions

- CASE 1: Both on the same side.

They were colored differently before.

(Why? Induction hypothesis.)

and are different now!

- CASE 2: On different sides of the  $n$ 'th line. Color of bottom reversed, so they are different.



Warning: Be careful when using induction!

Let  $x, y$  be integers and  $x, y \geq 0$ .

Let

$$\max(x, y) = \begin{cases} x & \text{if } x \geq y \\ y & \text{otherwise} \end{cases}$$

What is **wrong** with the proof of the following theorem?

**Theorem.** If  $\max(x, y) = n$ , then  $x = y$   
(this implies if  $x, y \geq 0$  and are integers, then  $x = y$ )

*Proof.* Induction base:  $n = 0$ . This is correct since when  $\max(x, y) = 0$  then  $x = y = 0$ .

Assume the result true for  $n = k$ , consider  $n = k + 1$ .

Let  $x_1 = x - 1$ ,  $y_1 = y - 1$ . Then  $\max(x_1, y_1) = k$

By the induction hypothesis,  $x_1 = y_1$ , therefore  $x = y$ .  $\square$

## Another wrong inductive proof

**Theorem.** All colours are the same.

*Proof.*

Induction base:  $n = 1$  - true (only one colour)

Induction hypothesis: Assume the statement true for  $n = k$ .

Consider  $n = k + 1$ .

We have  $c_1, c_2, \dots, c_{k+1}$ .

By the induction hypothesis,  $c_1, c_2, \dots, c_k$  are the same and  $c_2, c_3, \dots, c_{k+1}$  are the same. Since two sets overlap,  $c_1, c_2, \dots, c_{k+1}$  are the same.  $\square$

What is wrong? When  $n = 2$ ?

## Design with induction – Insertion Sort

- Array  $[1, \dots, n]$  of  $n$  elements
- Goal:  $A[1]$  is the smallest,  $A[2]$  is the second smallest and so on.
- Assume  $A[1, \dots, i]$  are sorted. It is then easy to sort  $A[1, \dots, i + 1]$  by inserting  $A[i + 1]$  into the list.

### Pseudocode INSERTION-SORT(A)

1. **for**  $j$  from 2 to  $\text{length}(A)$  do
2.    $key \leftarrow A[j]$
3.   //Insertion
4.    $i \leftarrow j - 1$
5.   **while**  $i > 0$  and  $A[i] > key$  do
6.      $A[i + 1] \leftarrow A[i]$
7.      $i \leftarrow i - 1$
8.    $A[i + 1] \leftarrow key$

- To insert  $A[i + 1]$  we need at most  $i$  comparisons and  $i + 1$  data movement.
- This implies  $T(n) \leq c \cdot n^2$  which means  $T(n) = O(n^2)$
- We can use induction to show the correctness of insertion sort.

At the start of each iteration of the **for** loop, the subarray  $A[1..j - 1]$  consists of the elements originally in  $A[1..j - 1]$  but in sorted order.

This can be considered as loop invariants or induction hypothesis.

For loop invariants, one has to check the initiation condition.

For induction, one has to check base case.

## Design with induction – Polynomial Evaluation

Given  $\bar{a} = (a_n, a_{n-1}, \dots, a_1, a_0)$  and  $x$

Compute  $p_n(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0$ .

### First Try

Hypothesis: we can compute  $p_{n-1}(x)$

Then:  $p_n(x) = a_n x^n + p_{n-1}(x)$

This works, but it isn't too clever.

Cost:  $n + (n - 1) + (n - 2) + \dots + 1 = n(n + 1)/2$  multiplications

$1 + 1 + \dots + 1 = n$  additions

## Stronger hypothesis

### Second try

Hypothesis: we know how to compute  $p_{n-1}(x)$  and  $x^{n-1}$ .

one multiplication to get  $x^n$  from  $x^{n-1}$

one multiplication to get  $a_n x^n$

one addition to get  $p_n(x)$

*Cost:*  $2n$  multiplications and  $n$  additions

This is better than  $O(n^2)$  multiplications.

### Third try (Reversed order)

Hypothesis: we know how to evaluate the polynomial with coefficients  $a_n, a_{n-1}, \dots, a_2, a_1$  at the point  $x$ , i.e.

$$p'_{n-1}(x) = a_n x^{n-1} + a_{n-1} x^{n-2} + \dots a_2 x + a_1$$

Then  $p_n(x) = x p'_{n-1}(x) + a_0$ .

*Cost:* 1 multiplication and 1 addition per term.

$n$  multiplications

$n$  additions

## Algorithm

$$a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0 =$$
$$((\dots ((a_n x + a_{n-1})x + a_{n-2})x + a_{n-3}) \dots )x + a_1)x + a_0$$

$p \leftarrow a_n$   
for  $i := 1$  to  $n$  do  
     $p \leftarrow x \cdot p + a_{n-i}$

Try variations on induction.

Exercise: Write the pseudocode of the other approaches.



## Other methods

Induction is not the only method to prove.

**Theorem.** Let  $G = (V, E)$  be an undirected graph. If each vertex of  $G$  has degree greater than one, then there is a cycle.

*Proof:* Consider  $P_{i_1}$ , choose one edge leaving  $P_{i_1}$ .

This edge leads us to  $P_{i_2}$ . We can choose another edge to leave  $P_{i_2}$ .

Repeat, we will have  $P_{i_1}, P_{i_2} \dots P_{i_n} \dots$

Since  $G$  has only  $|V|$  vertices, there must be  $P_{i_s}$  and  $P_{i_t}$  such that  $s < t$  and  $P_{i_s} = P_{i_t}$ .

This is a cycle.  $\square$

How to prove this theorem using induction?