Induction and Recursion

Jorge A. Cobb
Based on the slides by Lucia Moura, U. of Ottawa

U. T. Dallas

February 20, 2019

Table of contents

What general proof method would you use here?

- **Summation formulas**
  Prove that $1 + 2 + 2^2 + \cdot + 2^n = 2^{n+1} - 1$, for all integers n, $n \geq 0$.
- **Inequalities**
  Prove that $2n < n!$, for every positive integer n, with $n \geq 4$.
- **Divisibility results**
  Prove that $n^3 - n$ is divisible by 3, for every positive integer n.
- **Results about sets**
  Prove that if S is a set with n elements where n is a nonnegative integer, then S has 2^n subsets.
- **Geometry**
  Show that for n a positive integer, every $2^n \times 2^n$ checkerboard with one square removed can be tiled using right triominoes (L shape).
- **Results about algorithms**
  Prove that procedure $\text{FAC}(n)$ returns $n!$ for all positive $n \geq 0$.

Mathematical Induction

Principle of Mathematical Induction

Suppose you want to prove that a statement about an integer n is true for every integer n, where $n \geq c$ for some integer constant c (c is usually 1).

- Define a predicate $P(n)$ that describes the statement to be proven about n.
- To prove that $P(n)$ is true for all $n \geq c$, do the following two steps:
  - *Basis Step:* Prove that $P(c)$ is true.
  - *Inductive Step:* Let $k \geq c$. Assume $P(k)$ is true, and prove that $P(k+1)$ is true.

Applying the Induction Principle

Prove that $2^n < n!$ for every positive integer n, with $n \geq 4$.

- What is $P(n)$? $P(n) = 2^n < n!$
- What is c? $c = 4$

Proof:

BASIS: Show that $P(4)$, i.e., $2^4 < 4!$, is true.

$$2^4 = 16 < 24 = 4 \cdot 3 \cdot 2 \cdot 1 = 4!$$

INDUCTIVE STEP: Let $k \geq 4$. We show that

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

$$\begin{aligned} & 2^{k+1} \\ = \; & 2^k \cdot 2 \\ < \; & k! \cdot 2 \text{ (from induction hypothesis)} \\ < \; & k! \cdot k \text{ (why??)} \\ = \; & (k+1)! \end{aligned}$$

□

Another example of proof by induction

Prove that $1 + 2 + 3 \cdots + n = \frac{n(n+1)}{2}$ for every positive integer n.

- What is $P(n)$? $P(n) \equiv (\sum_{j=1}^{n} j = \frac{n(n+1)}{2})$
- What is c? $c = 1$

Proof:

BASIS: Show that $P(1)$, i.e., $(\sum_{j=1}^{1} j = \frac{1(1+1)}{2})$ is true.

trivial

INDUCTIVE STEP: Let $k \geq 1$.We show that

$$\left(\sum_{j=1}^{k} j = \frac{k(k+1)}{2}\right) \rightarrow \left(\sum_{j=1}^{k+1} j = \frac{(k+1)(k+2)}{2}\right)$$

proof continued ...

$$
\begin{aligned}
& \sum_{j=1}^{k+1} j \\
= \quad & k+1+\sum_{j=1}^{k} j \\
= \quad & k+1+\frac{k(k+1)}{2} \text{ (from induction hypothesis)} \\
= \quad & \frac{2(k+1)+k(k+1)}{2} \\
= \quad & \frac{(k+1)(k+2)}{2} \text{ as desired}
\end{aligned}
$$

A final example using numbers

Prove that $3|(n^3 - n)$ for every positive integer n.

- What is $P(n)$? $P(n) \equiv 3|(n^3 - n)$
- What is c? $c = 1$

Proof:

BASIS: Show that $P(1)$, i.e., $3|(1 - 1)$ is true.

i.e., $3|(0 - 0)$ which is trivial

INDUCTIVE STEP: Let $k \geq 1$. We show that

$$\big(3|(k^3 - k)\big) \rightarrow \big(3|((k + 1)^3 - (k + 1))\big)$$

proof continued ...

$$\begin{aligned} & (k+1)^3 - (k+1) \\ = \; & k^3 + 3k^2 + 3k + 1 - k - 1 \\ = \; & (k^3 - k) + (3k^2 + 3k) \end{aligned}$$

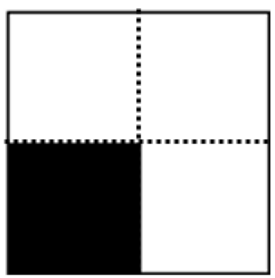
Both of these terms are divisible by three. Why??

- The first one from the induction hypothesis
- and the second by factoring 3.

Example not using algebra
"Tiling" with L-shaped pieces

- Any 2^n x 2^n checkered board with one 1x1 missing square can be covered with triominoes (L-shaped 3-squares)

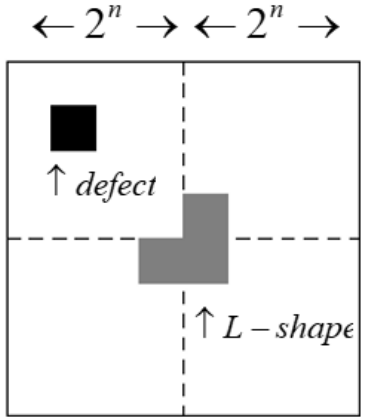
- Basis: Consider the following "defective", one missing square, of size 2^n, n=1



- Notice that when $n = 1$, the entire board can be tiled with a single *L-shaped* piece

continued …

- Inductive step
- Consider now the following defective $2^{n+1}x2^{n+1}$ board



- Divide the board into 4 equal pieces each of size 2^n
- Place a "starting" *L-shape* in the center of the board
- Since it has a tile already, you can't retile over it
- Hence, each 2^nx2^n section is defective in one tile
- By induction hypothesis, each of the four sections (each with one defect can be tiled).

Strong Induction

Principle of Strong Induction

Suppose you want to prove that a statement about an integer n is true for every integer n, where $n \geq c$ for some integer constant c (c is often just 1).

- Define a predicate $P(n)$ that describes the statement to be proven about n.
- To prove that $P(n)$ is true for all n, $n \geq c$, do the following two steps:
  - *Basis Step*: Prove that $P(c)$ is true.
  - *Inductive Step*: Let $k \geq c$. Assume $P(c), P(c+1), \ldots, P(k)$ are all true, and prove that $P(k+1)$ is true.

Use strong induction to prove:

Theorem

(The Fundamental Theorem of Arithmetic) Every positive integer greater than 1 can be written uniquely as a prime or as the product of two or more primes where the prime factors are written in order of nondecreasing size.

Proof:

Part 1: Every positive integer greater than 1 can be written as a prime or as the product of two or more primes.

Part 2: Show uniqueness, when the primes are written in nondecreasing order.

Proof of Part 1: Consider $P(n)$ to be "n can be written as a prime or as the product of two or more primes." We will use strong induction to show that $P(n)$ is true for every integer $n \geq 2$.

BASIS STEP: $P(2)$ is true, since 2 can be written as a prime, itself.

INDUCTION STEP: Let $k \geq 2$. Assume $P(2), P(3), \ldots, P(k)$ are true. We will prove that $P(k+1)$ is true, i.e., that $k+1$ can be written as a prime or the product of two or more primes.

Case 1: $k+1$ is prime.

If $k+1$ is prime, then the statement is true since $k+1$ can be written as a prime, i.e., itself.

Case 2: $k+1$ is composite (a product of two or more numbers).

By definition, there exist two positive integers a and b with $2 \leq a \leq b < k+1$, such that $ab = k+1$. Since $a, b < k+1$, we know by induction hypothesis that a and b can each be written as a prime or the product of two or more primes. Thus, $k+1 = ab$ can be written as a product of two or more primes, namely those primes in the prime factorization of a and those primes in the prime factorization of b.

We want to prove Part 2. The following Lemma has been proven.

Lemma (A)

If a, b, and c are positive integers such that gcd$(a, b) = 1$ and $a|bc$, then $a|c$.

We prove the following lemma using induction.

Lemma (B)

If p is a prime and $p|a_1a_2 \cdots a_n$, where each a_i is an integer and $n \geq 1$, then $p|a_i$ for some i, $1 \leq i \leq n$.

Proof: Let $P(n)$ be the statement "If $a_1, a_2, \ldots, a_n$ are integers and p is a prime number such that $p|a_1a_2 \cdots a_n$, then $p|a_i$ for some i, $1 \leq i \leq n$". We will prove $P(n)$ is true for all $n \geq 1$.

Let $P(n)$ be the statement "If $a_1, a_2, \ldots, a_n$ are integers and p is a prime number such that $p|a_1 a_2 \cdots a_n$, then $p|a_i$ for some i, $1 \leq i \leq n$".
We will prove $P(n)$ is true for all $n \geq 1$.

Basis: Prove that $P(1)$ is true.
The statement is trivially true, since for $n = 1$, $p|a_1$ already gives that $p|a_i$ for some i, $1 \leq i \leq 1$.

Induction step: Let $n \geq 2$. Assume $P(n-1)$ is true. Prove $P(n)$ is true.
Let p be a prime such that $p|a_1 a_2 \cdots a_n$. In the case that $p|a_n$, we are done. So, consider the case $p \nmid a_n$. Since p is prime, we have $\gcd(p, a_n) = 1$, thus, by Lemma A, $p|a_1 \ldots a_{n-1}$. By induction hypothesis, we have that $p|a_i$ for some i, $1 \leq i \leq n-1$. Combining both cases, we get that $p|a_i$ for some i, $1 \leq i \leq n$.
□

Proof of Part 2: (uniqueness of the prime factorization of a positive integer).

Suppose by contradiction that n can be written as a product of primes in two different ways, say $n = p_1 p_2 \dots p_s$ and $n = q_1 q_2 \dots q_t$, where each p_i and q_j are primes such that $p_1 \leq p_2 \leq \cdots \leq p_s$ and $q_1 \leq q_2 \leq \cdots \leq q_t$.
When we remove all common primes from the two factorizations, we have: $p_{i_1} p_{i_2} \cdots p_{i_u} = q_{j_1} q_{j_2} \cdots q_{j_v}$, where no primes occur on both sides of this equations and u and v are positive integers.
By Lemma B, p_{i_1} must divide q_{j_k} for some k, $1 \leq k \leq v$. Since p_{i_1} and q_{j_k} are primes we must have $p_{i_1} = q_{j_k}$, which contradicts the fact that no primes appear on both sides of the given equation.
□

Strong Induction Generalized

Principle of Generalized Strong Induction

Suppose you want to prove that a statement about an integer n is true for every integer n, where $n \geq c$ for some integer constant c (c is often just 1). Let j be a non-negative integer.

- Define a predicate $P(n)$ that describes the statement to be proven about n.
- To prove that $P(n)$ is true for all n, $n \geq c$, do the following two steps:
  - *Basis Step*: Prove that each of $P(c), P(c+1), P(c+2), \ldots, P(c+j)$ is true.
  - *Inductive Step*: Let $k \geq c+j$. Assume $P(c), P(c+1), P(c+2), \ldots, P(k)$ are all true, and prove that $P(k+1)$ is true.

Example of Strong Induction

Prove that every amount of postage 12 cents or more can be formed using just 4-cent and 5-cent stamps.

- What is $P(n)$? $P(n) = n$ cents can be formed from 4-cent and 5-cent stamps.
- What is c? $c = 12$
- What is j? Not clear from the statement of the problem. Below, we will replace a 4-cent stamp by a 5 cent stamp, so let's choose $j = 3$.

Proof:

Basis: Show that $P(12), P(13), P(14), P(15)$ are all true (on your own).

Inductive Step: Let $k \geq 15$. We show that

$$(P(12) \wedge P(13) \wedge P(14) \wedge P(15) \wedge \ldots \wedge P(k)) \rightarrow P(k+1)$$

$k+1$ can be done from the stamps chosen for $k-3$ plus a 4 cent stamp.

Aha! We know that $P(k-3)$ is true because of ind. hyp. and $12 \leq k-3$ (recall that $k \geq 15$).

We are done!

Recursive Definitions

We can use recursion to define:

- functions,
- sequences,
- sets.

Mathematical induction and strong induction can be used to prove results about recursively defined sequences and functions.

Structural induction is used to prove results about recursively defined sets.

Recursively Defined Functions

Examples:

- Defining the factorial function recursively:

$$\begin{aligned} F(0) &= 1, \\ F(n) &= n \times F(n-1), \text{ for } n \geq 1. \end{aligned}$$

- Defining the maximum number of comparisons for the Mergesort algorithm (given in page 318):

$$\begin{aligned} T(1) &= 0, \\ T(n) &= T(\lfloor n/2 \rfloor) + T(\lceil n/2 \rceil) + n - 1, \text{ for } n \geq 2. \end{aligned}$$

- Number of moves needed to solve the Hanoi tower problem:

$$\begin{aligned} H(1) &= 1, \\ H(n) &= 2H(n-1) + 1, \text{ for } n \geq 2. \end{aligned}$$

Recursively Defined Sequences

Consider the Fibonacci numbers, recursively defined by:

$$\begin{aligned} f_0 &= 0, \\ f_1 &= 1, \\ f_n &= f_{n-1} + f_{n-2}, \text{ for } n \geq 2. \end{aligned}$$

Prove that whenever $n \geq 3$, $f_n > \alpha^{n-2}$ where $\alpha = (1 + \sqrt{5})/2$.

Let $P(n)$ be the statement "$f_n > \alpha^{n-2}$". We will show that $P(n)$ is true for $n \geq 3$ using strong induction.

BASIS: We show that $P(3)$ and $P(4)$ are true:
$\alpha = (1+\sqrt{5})/2 < (1+3)/2 = 2 = f_3$.
$\alpha^2 = ((1+\sqrt{5})/2)^2 = (1^2+2\sqrt{5}+5)/4 = (3+\sqrt{5})/2 < (3+3)/2 = 3 = f_4$.

INDUCTIVE STEP: Let $k \geq 4$. Assume $P(j)$ is true for all integers j with $3 \leq j \leq k$. Prove that $P(k+1)$ is true.
We have:

$$\begin{aligned} f_{k+1} &= f_k + f_{k-1}, \quad \text{(by the definition of the Fibonacci sequence)} \\ &> \alpha^{k-2} + \alpha^{k-3}, \quad \text{(by induction hypothesis)} \\ &= \alpha^{k-3}(\alpha+1) = \alpha^{k-3}((1+\sqrt{5})/2+1) = \alpha^{k-3}((3+\sqrt{5})/2) \\ &= \alpha^{k-3}\alpha^2 = \alpha^{k-1}. \end{aligned}$$

Recursively Defined Sets and Structures

Definition (Set of strings over an alphabet)

The set Σ^* of strings over the alphabet Σ can be defined recursively by:
BASIS STEP: $\lambda \in \Sigma^*$ (where λ is the empty string)
RECURSIVE STEP: If $w \in \Sigma^*$ and $x \in \Sigma$, then $wx \in \Sigma^*$.

Example: If $\Sigma = \{0, 1\}$, then
$\Sigma^* = \{\lambda, 0, 1, 00, 01, 10, 11, 000, 001, 010, 011, \ldots\}$.

Definition (Well-formed formulas of Operators and Operands)

BASIS STEP: x is a well-formed formula if x is a numeral or variable.
RECURSIVE STEP: If F and G are well-formed formulas, then $(F + G)$, $(F - G)$, $(F * G)$, (F/G) and $(F \uparrow G)$ are well-formed formulas.

Example: The following are well-formed formulas:
$(x * 3)$, $(3/0)$, $((x + 2) * y)$, $((2 + 3) - (x/y))$, etc.

Structural Induction

Structural induction is used to show results about recursively defined sets.

Principle (of Structural Induction)

To show that a statement holds for all elements of a recursively defined set, use the following steps:

- BASIS STEP: *Prove that the statement holds for all elements specified in the basis step of the set definition.*
- RECURSIVE STEP: *Prove that if the statement is true for each of the elements used to construct elements in the recursive step of the set definition, then the result holds for these new elements.*

The validity of this principle comes from the validity of mathematical induction, as we can transform the above argument on an induction on n where n is the number of applications of the recursive step of the set definition needed to obtain the element we are analysing.

Example of Structural Induction I

Prove that every well-formed formula of Operators and Operands contains an equal number of left and right parentheses.

Proof by structural induction:

BASIS STEP: A numeral or a variable, each contains 0 parentheses, so clearly they contain an equal number of right and left parentheses.

RECURSIVE STEP: Assume F and G are well-formed formulas each containing an equal number of left and right parentheses. That is, if l_F and l_G are the number of left parentheses in F and G, respectively, and r_F and r_G are the number of right parentheses in F and G, respectively, then $l_F = r_F$ and $l_G = r_G$. We need to show that $(F + G)$, $(F - G)$, $(F * G)$, (F/G) and $(F \uparrow G)$ also contain an equal number of left and right parenthesis. For each of these well-formed formulas, the number of left parentheses is $L = l_F + l_G + 1$ and the number of right parentheses is $R = r_F + r_G + 1$. Since $l_F = r_F$ and $l_G = r_G$, it follows that $L = l_F + l_G + 1 = r_F + r_G + 1 = R$. This concludes the inductive proof. $\square$

Example of Structural Induction II

Recall the definition of a set of strings.

Definition (Set of strings over an alphabet)

The set Σ^* of strings over the alphabet Σ can be defined recursively by:
BASIS STEP: $\lambda \in \Sigma^*$ (where λ is the empty string)
RECURSIVE STEP: If $w \in \Sigma^*$ and $x \in \Sigma$, then $wx \in \Sigma^*$.

We now give a definition of concatenation of two strings.
Note how this definition is built on the definition of string.

Definition (Concatenation of two strings)

BASIS STEP: If $w \in \Sigma^*$, then $w \cdot \lambda = w$.
RECURSIVE STEP: If $w_1 \in \Sigma^*$, $w_2 \in \Sigma^*$ and $x \in \Sigma$, then
$w_1 \cdot (w_2 x) = (w_1 \cdot w_2)x$.

We now give a recursive definition of the reversal of a string.

Definition (Reversal of a string)

BASIS STEP: $\lambda^R = \lambda$
RECURSIVE STEP: If $w \in \Sigma^*$ and $x \in \Sigma$, then $(wx)^R = x \cdot (w)^R$.

Exercise: Use structural induction to prove that if w_1 and w_2 are strings, then $(w_1 \cdot w_2)^R = w_2^R \cdot w_1^R$.

Note that this proof needs to use the 3 definitions given above.

Proving the correctness of recursive programs

Mathematical induction (and strong induction) can be used to prove that a recursive algorithm is correct:

to prove that the algorithm produces the desired output for all possible input values.

We will see some examples next.

Recursive algorithm for computing a^n

procedure power(a: nonzero real number, n: nonnegative integer)
  if $(n = 0)$ **then** return 1
    else return $a \times$power$(a, n - 1)$

We will prove by mathematical induction on n that the algorithm above is correct.
We will show $P(n)$ is true for all $n \geq 0$, for
$P(n)$: For all nonzero real numbers a, power(a, n) correctly computes a^n.

Proving $power(a, n)$ is correct

Basis: If $n = 0$, the first step of the algorithm tells us that power(a,0)=1. This is correct because $a^0 = 1$ for every nonzero real number a, so $P(0)$ is true.

Inductive step:
Let $k \geq 0$.
Inductive hypothesis: power$(a, k) = a^k$, for all $a \neq 0$.
We must show next that power$(a, k + 1) = a^{k+1}$.
Since $k + 1 > 0$ the algorithm sets power$(a, k + 1) = a \times$power(a, k).
By inductive hypotheses $power(a, k) = a^k$, so
power$(a, k + 1) = a \times$power$(a, k) = a \times a^k = a^{k+1}$.

Recursive algorithm for computing $b^n \mod m$

procedure mpower(b, n, m: integers with $m \geq 2, n \geq 0$)
  if $n = 0$ then return 1;
  else if n is even then return mpower$(b, n/2, m)^2 \mod m$
  else return ((mpower$(b, \lfloor n/2 \rfloor, m)^2 \mod m) * (b \mod m)) \mod m$

Examples:

$$
\begin{aligned}
& power(2,5,6) = \\
= \; & ((power(2,2,6)^2 \mod 6) * (2 \mod 6)) \mod 6 \\
= \; & (((power(2,1,6)^2 \mod 6)^2 \mod 6) * (2)) \mod 6 \\
= \; & ((((power(2,0,6)^2 \mod 6) * (2 \mod 6)) \mod 6)^2 \mod 6)^2 \mod 6 \\
& * 2) \mod 6 \\
= \; & ((((1^2 \mod 6) * 2) \mod 6)^2 \mod 6)^2 \mod 6) * 2) \mod 6 \\
= \; & 2
\end{aligned}
$$

Proving $mpower(a, n, m)$ is correct, using induction on n

Basis: Let b and m be integers with $m \geq 2$, and $n = 0$. In this case, the algorithm returns 1. This is correct because $b^0 \mod m = 1$.

Inductive step:

Induction hypothesis: Let $k \geq 1$. Assume power$(b, j, m) = b^j \mod m$ for all integers j with $0 \leq j \leq k-1$, whenever b is a positive integer and m is an integer with $m \geq 2$.

We must show next that power$(b, k, m) = b^k \mod m$. There are two cases to consider.

- Case 1: k is even. In this case, the algorithm returns mpower$(b, k/2, m)^2 \mod m = (i.h.)(b^{k/2} \mod m)^2 \mod m = b^k \mod m$.
- Case 2: k is odd. In this case, the algorithm returns
  $((\text{mpower}(b, \lfloor k/2 \rfloor, m)^2 \mod m) * (b \mod m)) \mod m$
  $= (i.h.)(b^{\lfloor k/2 \rfloor} \mod m)^2 \mod m) * (b \mod m)) \mod m$
  $= (b^{2\lfloor k/2 \rfloor + 1} \mod m) = b^k \mod m$.