Review of Discrete Math

Xiao Zhang

2020-12-17

Chapter 5： Induction and recursion

- Mathematical Induction
- Strong Induction
- Well-Ordering
- Recursive Definitions
- Structural Induction
- Recursive Algorithms

Proof

Application

Mathematical Induction

Principle of Mathematical Induction: To prove that $P(n)$ is true for all positive integers n, we complete these steps:

- *Basis Step*: Show that $P(1)$ is true.
- *Inductive Step*: Show that $P(k) \to P(k+1)$ is true for all positive integers k.

To complete the inductive step, assuming the *inductive hypothesis* that $P(k)$ holds for an arbitrary integer k, show that must $P(k+1)$ be true.

Mathematical induction can be expressed as the rule of inference

$$(P(1) \land \forall k\,(P(k) \to P(k+1))) \to \forall n\, P(n),$$

Example: Use mathematical induction to prove that $n^3 - n$ is divisible by 3, for every positive integer n.

Solution: Let $P(n)$ be the proposition that $n^3 - n$ is divisible by 3.

- BASIS STEP: $P(1)$ is true since $1^3 - 1 = 0$, which is divisible by 3.
- INDUCTIVE STEP: Assume $P(k)$ holds, i.e., $k^3 - k$ is divisible by 3, for an arbitrary positive integer k. To show that $P(k+1)$ follows:

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

By the inductive hypothesis, the first term $(k^3 - k)$ is divisible by 3 and the second term is divisible by 3 since it is an integer multiplied by 3. So by part (i) of Theorem 1 in Section 4.1 , $(k+1)^3 - (k+1)$ is divisible by 3.

Therefore, $n^3 - n$ is divisible by 3, for every integer positive integer n.

Template for Proofs by Mathematical Induction

1. Express the statement that is to be proved in the form "for all $n \geq b$, $P(n)$" for a fixed integer b.
2. Write out the words "Basis Step." Then show that $P(b)$ is true, taking care that the correct value of b is used. This completes the first part of the proof.
3. Write out the words "Inductive Step."
4. State, and clearly identify, the inductive hypothesis, in the form "assume that $P(k)$ is true for an arbitrary fixed integer $k \geq b$."
5. State what needs to be proved under the assumption that the inductive hypothesis is true. That is, write out what $P(k+1)$ says.
6. Prove the statement $P(k+1)$ making use the assumption $P(k)$. Be sure that your proof is valid for all integers k with $k \geq b$, taking care that the proof works for small values of k, including $k = b$.
7. Clearly identify the conclusion of the inductive step, such as by saying "this completes the inductive step."
8. After completing the basis step and the inductive step, state the conclusion, namely that by mathematical induction, $P(n)$ is true for all integers n with $n \geq b$.

Strong Induction

Strong Induction: To prove that $P(n)$ is true for all positive integers n, where $P(n)$ is a propositional function, complete two steps:

- *Basis Step*: Verify that the proposition $P(1)$ is true.
- *Inductive Step*: Show the conditional statement $[P(1) \wedge P(2) \wedge \cdots \wedge P(k)] \rightarrow P(k + 1)$ holds for all positive integers k.

We can always use strong induction instead of mathematical induction. But there is no reason to use it if it is simpler to use mathematical induction. (*See page* 335 *of text*.)

In fact, the principles of mathematical induction, strong induction, and the well-ordering property are all equivalent. (*Exercises* 41-43)

Sometimes it is clear how to proceed using one of the three methods, but not the other two.

Example: Show that if n is an integer greater than 1, then n can be written as the product of primes.

Solution: Let $P(n)$ be the proposition that n can be written as a product of primes.

- BASIS STEP: $P(2)$ is true since 2 itself is prime.
- INDUCTIVE STEP: The inductive hypothesis is $P(j)$ is true for all integers j with $2 \leq j \leq k$. To show that $P(k + 1)$ must be true under this assumption, two cases need to be considered:
  - If $k + 1$ is prime, then $P(k + 1)$ is true.
  - Otherwise, $k + 1$ is composite and can be written as the product of two positive integers a and b with $2 \leq a \leq b < k + 1$. By the inductive hypothesis a and b can be written as the product of primes and therefore $k + 1$ can also be written as the product of those primes.

Hence, it has been shown that every integer greater than 1 can be written as the product of primes. ◀

Well-Ordering

Well-ordering property: Every nonempty set of nonnegative integers has a least element.

The well-ordering property is one of the axioms of the positive integers listed in Appendix 1.

The well-ordering property can be used directly in proofs, as the next example illustrates.

The well-ordering property can be generalized.

- **Definition:** A set is *well ordered if every subset has a least element*.
  - **N** is well ordered under ≤.
  - The set of finite strings over an alphabet using lexicographic ordering is well ordered.

Example: Use the well-ordering property to prove the division algorithm, which states that if a is an integer and d is a positive integer, then there are unique integers q and r with $0 \leq r < d$, such that $a = dq + r$.

Solution: Let S be the set of nonnegative integers of the form $a - dq$, where q is an integer. The set is nonempty since $-dq$ can be made as large as needed.

- By the well-ordering property, S has a least element $r = a - dq_0$. The integer r is nonnegative. It also must be the case that $r < d$. If it were not, then there would be a smaller nonnegative element in S, namely, $a - d(q_0 + 1) = a - dq_0 - d = r - d > 0$.
- Therefore, there are integers q and r with $0 \leq r < d$.

(*uniqueness of q and r is Exercise* 37)

Recursive Definitions and Recusive difinition

Definition: A *recursive* or *inductive definition* of a function consists of two steps.

- BASIS STEP: Specify the value of the function at zero.
- RECURSIVE STEP: Give a rule for finding its value at an integer from its values at smaller integers.

A function $f(n)$ is the same as a sequence a_0, a_1, … , where a_i, where $f(i) = a_i$. This was done using recurrence relations in Section 2.4.

Recursive definitions of sets have two parts:

- The *basis step* specifies an initial collection of elements.
- The *recursive step* gives the rules for forming new elements in the set from those already known to be in the set.

Sometimes the recursive definition has an ***exclusion rule***, which specifies that the set contains nothing other than those elements specified in the basis step and generated by applications of the rules in the recursive step.

We will always assume that the exclusion rule holds, even if it is not explicitly mentioned.

We will later develop a form of induction, called *structural induction*, to prove results about recursively defined sets.

Example: Suppose f is defined by:

$f(0) = 3$,

$f(n + 1) = 2f(n) + 3$

Find $f(1)$, $f(2)$, $f(3)$, $f(4)$

Solution:

- $f(1) = 2f(0) + 3 = 2\cdot3 + 3 = 9$
- $f(2) = 2f(1)+ 3 = 2\cdot9 + 3 = 21$
- $f(3) = 2f(2) + 3 = 2\cdot21 + 3 = 45$
- $f(4) = 2f(3) + 3 = 2\cdot45 + 3 = 93$

Example : Subset of Integers S:

BASIS STEP: $3 \in S$.

RECURSIVE STEP: If $x \in S$ and $y \in S$, then $x + y$ is in S.

- Initially 3 is in S, then $3 + 3 = 6$, then $3 + 6 = 9$, etc.

Binary tree、 Rooted tree

Example: Give a recursive definition of the set of balanced parentheses(圆括号) P.

Solution:

BASIS STEP: $() \in P$

RECURSIVE STEP: If $w \in P$, then $()\,w \in P$, $(w) \in P$ and $w\,() \in P$.

Show that (() ()) is in P.

Why is))(() not in P?

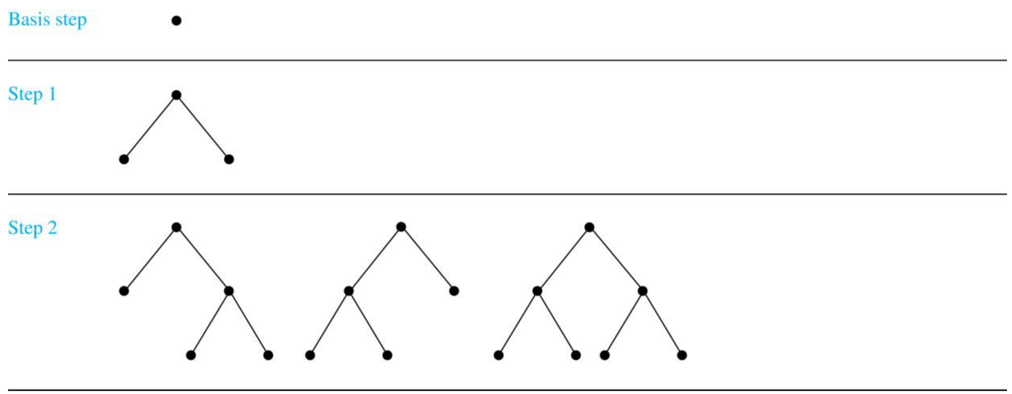
Structural Induction

Definition: To prove a property of the elements of a recursively defined set, we use *structural induction*.

BASIS STEP: Show that the result holds for all elements specified in the basis step of the recursive definition.

RECURSIVE STEP: Show that if the statement is true for each of the elements used to construct new elements in the recursive step of the definition, the result holds for these new elements.

The validity of structural induction can be shown to follow from the principle of mathematical induction.



Definition: The *height* $h(T)$ of a full binary tree T is defined recursively as follows:

- BASIS STEP: The height of a full binary tree T consisting of only a root r is $h(T) = 0$.
- RECURSIVE STEP: If T_1 and T_2 are full binary trees, then the full binary tree $T = T_1 \cdot T_2$ has height $h(T) = 1 + \max(h(T_1), h(T_2))$.

The number of vertices $n(T)$ of a full binary tree T satisfies the following recursive formula:

- **BASIS STEP**: The number of vertices of a full binary tree T consisting of only a root r is $n(T) = 1$.
- **RECURSIVE STEP**: If T_1 and T_2 are full binary trees, then the full binary tree $T = T_1 \cdot T_2$ has the number of vertices
$$n(T) = 1 + n(T_1) + n(T_2).$$

Recursive Algorithms

Definition: An algorithm is called *recursive* if it solves a problem by reducing it to an instance of the same problem with smaller input.

For the algorithm to terminate, the instance of the problem must eventually be reduced to some initial case for which the solution is known.

Exercise

- §5.1 6,10,22
- §5.2 2,8,14,28
- §5.3 4,10,18
- §5.4 2,16,24,28