CP214: Discrete Structures
Induction and Recursive Programs

Dr. Zia Ud Din

Induction

Induction

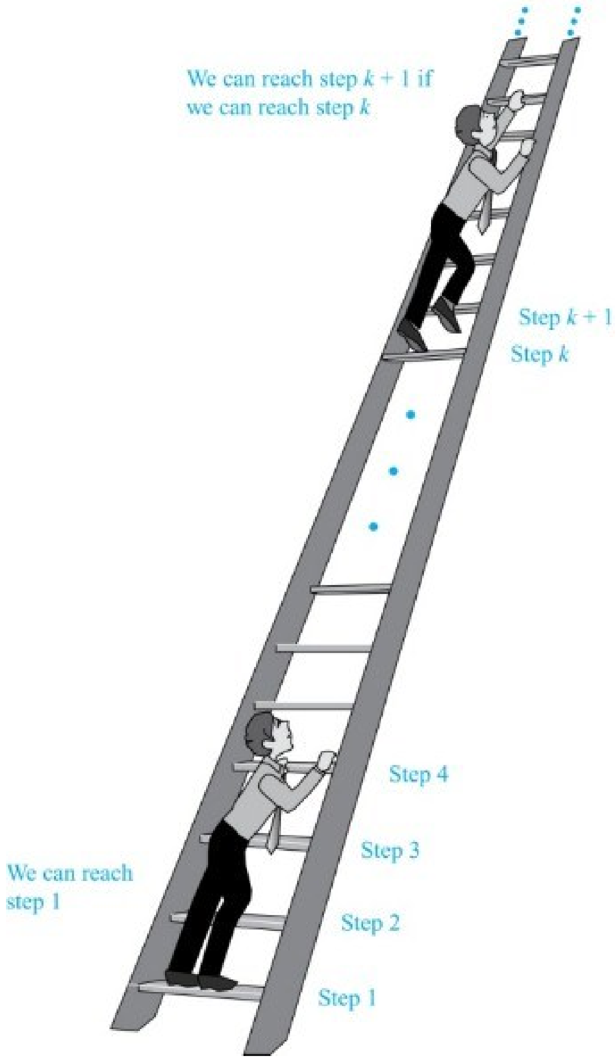
- The **principle of mathematical induction** is a useful tool for proving that a certain predicate is true for **all positive integers**.

Climbing an Infinite Ladder

- Suppose we have an infinite ladder:

1. We can reach the first rung of the ladder.
2. If we can reach a particular rung of the ladder, then we can reach the next rung.

- From (1), we can reach the first rung. Then by applying (2), we can reach the second rung. Applying (2) again, the third rung. And so on. We can apply (2) any number of times to reach any particular rung, no matter how high up.
- This example motivates proof by mathematical induction.



Induction

- If we have a propositional function $P(n)$, and we want to prove that $P(n)$ is true for all positive integers n, we do the following:

- Show that $P(1)$ is true.
  (basis step)
- Show that if $P(k)$ then $P(k + 1)$ for any $k \in Z^+$.
  (inductive step)
- Then $P(n)$ must be true for any $n \in Z^+$.
  (conclusion)

Induction

- **Example:**

  Show that $n < 2^n$ for all positive integers n.

- **Solution:**

  Let $P(n)$ be the proposition "$n < 2^n$"

  1. Show that $P(1)$ is true.

  (basis step)

  $P(1)$ is true, because $1 < 2^1 = 2$.

Induction

2. Show that if $P(k)$ is true, then $P(k + 1)$ is true.
(inductive step)

Assume that $k < 2^k$ is true.

We need to show that $P(k + 1)$ is true, i.e. $k + 1 < 2^{k+1}$

We start from $k < 2^k$:

$k + 1 < 2^k + 1 < 2^k + 2^k = 2^{k+1}$

Therefore, if $k < 2^n$ then $k + 1 < 2^{k+1}$

Induction

Then P(n) must be true for any positive integer. (conclusion)

$n < 2^n$ is true for any positive integer.

End of proof.

Induction

- **Another Example ("Gauss"):**

  $1 + 2 + \ldots + n = n(n+1)/2$

- **Solution:**

  1. Show that P(1) is true.
     (basis step)

  For n = 1 we get 1 = 1 (1 + 1)/2 = 1. True.

Induction

2. Show that if $P(k)$ then $P(k + 1)$ for any $k \in Z^+$. (inductive step)

$$1 + 2 + \ldots + k = k\,(k + 1)/2$$

$$\begin{aligned} 1 + 2 + \ldots + k + (k + 1) &= k\,(k + 1)/2 + (k + 1) \\ &= (k + 1)\,(k/2 + 1) \\ &= (k + 1)\,(k + 2)/2 \\ &= (k + 1)\,((k + 1) + 1)/2 \end{aligned}$$

Induction

3. Then $P(n)$ must be true for any $n \in Z^+$. (conclusion)

$1 + 2 + \ldots + n = n(n + 1)/2$ is true for all $n \in Z^+$.

End of proof.

Practice Problems

1. Show that $1+3+5\ldots+(2n-1) = n^2$, where n is a positive integer.

2. Use mathematical induction to prove that $2^n < n!$, for every integer $n \geq 4$.

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

Guidelines: Mathematical Induction Proofs

- ***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, by mathematical induction, $P(n)$ is true for all integers n with $n \geq b$.

Strong Induction

- The second principle of mathematical induction:

- Show that $P(1)$ is true.
  (basis step)
- Show that if $P(1) \wedge P(2) \wedge \ldots \wedge P(k)$,
  then $P(k + 1)$ for any $k \in Z^+$.
  (inductive step)
- Then $P(n)$ must be true for any $n \in Z^+$.
  (conclusion)

Strong Induction

- **Example:**

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

- **Solution:**

- Show that P(2) is true.
  (basis step)

  2 is the product of one prime: itself.

Strong Induction

- Show that if $P(2) \land P(3) \land \ldots \land P(k)$,
  then $P(k + 1)$ for any $k \in Z^+$. (inductive step)

Two possible cases:

- If $(k + 1)$ is **prime**, then obviously $P(k + 1)$ is true.
- If $(k + 1)$ is **composite**, it can be written as the product of two integers a and b such that
  $2 \leq a \leq b < k + 1$.
- By the **induction hypothesis**, both a and b can be written as the product of primes.
- Therefore, $k + 1 = a \cdot b$ can be written as the product of primes.

Strong Induction

- Then P(n) must be true for any $n \in Z^+$.
  (conclusion)

We have shown that **every integer greater than 1** can be written as the product of primes.

End of proof.

Recursive Programs

Recursive Definitions

- **Recursion** is a principle closely related to mathematical induction.
- In a **recursive definition**, an object is defined in terms of itself.
- We can recursively define **sequences**, **functions** and **sets**.

Recursively Defined Functions

- **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 $a_0, a_1, \ldots$, where a_i, where $f(i) = a_i$.

Recursively Defined Functions

- **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:**

  $f(1) = 2f(0) + 3 = 2 \cdot 3 + 3 = 9$

  $f(2) = 2f(1) + 3 = 2 \cdot 9 + 3 = 21$

  $f(3) = 2f(2) + 3 = 2 \cdot 21 + 3 = 45$

  $f(4) = 2f(3) + 3 = 2 \cdot 45 + 3 = 93$

Recursively Defined Functions

- **How can we recursively define the factorial function $f(n) = n!$?**

$f(0) = 1$
$f(n + 1) = (n + 1)f(n)$

- **Solution:**

$f(1) = 1f(0) = 1 \cdot 1 = 1$
$f(2) = 2f(1) = 2 \cdot 1 = 2$
$f(3) = 3f(2) = 3 \cdot 2 = 6$
$f(4) = 4f(3) = 4 \cdot 6 = 24$

Recursively Defined Functions

- **Example: The Fibonacci numbers**

  $f(0) = 0, f(1) = 1$

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

- **Solution:**

  $f(1) = 1$

  $f(2) = f(1) + f(0) = 1 + 0 = 1$

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

  $f(4) = f(3) + f(2) = 2 + 1 = 3$

  $f(5) = f(4) + f(3) = 3 + 2 = 5$

  $f(6) = f(5) + f(4) = 5 + 3 = 8$

Recursively Defined Sets

- If we want to recursively define a set, we need to provide two things:
  - an **initial set** of elements (basis step),
  - **rules** for the construction of **additional** elements from elements already in the set (recursive step).

- **Example:** Let S be recursively defined by:

  $3 \in S$

  $(x + y) \in S$ if $(x \in S)$ and $(y \in S)$

- **Solution:** S is the set of positive integers divisible by 3.

Recursive Algorithms

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

- **Example:** Recursive Euclidean Algorithm

procedure gcd(a, b: nonnegative integers with a < b)
if a = 0 **then return** b
else return gcd(b **mod** a, a)

Recursive Algorithms

- **Example:** Recursive Fibonacci Algorithm

```
procedure fibo(n: nonnegative integer)
if n = 0 then return 0
else if n = 1 then return 1
else return fibo(n – 1) + fibo(n – 2)
```

Practice Problems

1. Give a recursive definition of the set of natural numbers **N**.
2. Give a recursive algorithm for computing $n!$, where n is a nonnegative integer.
3. Give a recursive algorithm for computing a^n, where a is a nonzero real number and n is a nonnegative integer.

Summary

- In this part, we discussed:
  - Induction
  - Recursive Definitions

References

- These slides are largely based on the following two sources:
  - Slides by Marc Pomplun, UMass Boston
  - Official McGraw Hill's Slides for our textbook