Mathematical Induction

What it is?
Why is it a legitimate proof method?
How to use it?

Some Logic Background

- **Implication**: A→B, where A and B are boolean values. If A, then B
  - If it rains today, I will use my umbrella.
    - When is this statement true?
      - When I use my umbrella
      - When it does not rain
    - When is it false?
      - Only when it rains and I do not use my umbrella.

A	B	A→B
T	T	T
T	F	F
F	T	T
F	F	T

Implication: A→B

A	B	A→B
T	T	T
T	F	F
F	T	T
F	F	T

- Conclusions:
  - If we know that **A→B** is true, and **B** is false, then **A** must be …
  - Another expression that is equivalent to A→B:
    - ¬ A OR B (think about the truth table for ¬A OR B)
  - What similar expression is equivalent to ¬(A→B)?

Contrapositive of A→B

A	B	A→B	¬B	¬A	¬B → ¬A
T	T	T	F	F	T
T	F	F	T	F	F
F	T	T	F	T	T
F	F	T	T	T	T

- ¬B → ¬A is called the **contrapositive** of A→B
- Notice that the third and sixth columns of the truth table are the same.
- Thus an implication is true if and only if its contrapositive is true.

Contradictions

- If A is a boolean value, the value of the expression A AND ¬A is _____. This expression is known as a **contradiction**.
- Putting this together with what we saw previously, if B → (A AND ¬A) **is True, what can we say about B?**
- **This is the basis for "proof by contradiction".**
  - **To show that B is true, we find an A for which we can show that ¬B → (A AND ¬A) is true.**
  - **This is the approach we will use in our proof that Mathematical induction works.**

The Well-ordering principle

- It's an axiom, not something that we can prove.
- WOP: Every non-empty set of non-negative integers has a smallest element.
- Note the importance of "non-empty", "non-negative", and "integers".
  - The empty set does not have a smallest element.
  - A set with no lower bound (such as the set of all integers) does not have a smallest element.
    - In the statement of WOP, we can replace "positive" with "has a lower bound"
  - Unlike integers, a set of rational numbers can have a lower bound but no smallest member.
- Assuming the well-ordering principle, we can prove that the principle of mathematical induction is true.

Recap: What kind of things do we try to prove *via* Mathematical Induction?

- In this course, it will be a property of positive integers (or non-negative integers, or integers larger than some specific number).
- A *property* p(n) is a boolean statement about the integer *n*. [p: int → boolean]
  - Example: p(n) could be "n is an even number".
  - Then p(4) is true, but p(3) is false.
- If we believe that some property p is true for **all** positive integers, induction gives us a way of proving it.

The Principle of Mathematical Induction

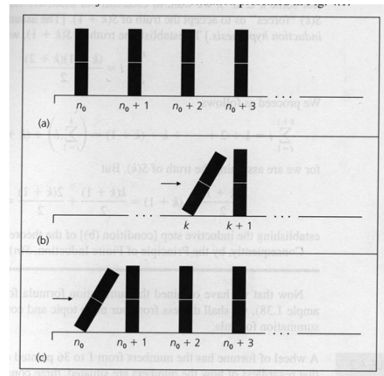
- To prove that p(n) is true for all n $\geq n_0$:
  - Show that $p(n_0)$ is true.
  - Show that for all k $\geq n_0$,

    p(k) implies p(k+1).

    I.e, show that whenever p(k) is true, then p(k+1) is true also.

Why does induction work? (Informal look)

- To prove that p(n) is true for all n $\geq n_0$:
  - Show that $p(n_0)$ is true.
  - Show that for all $k \geq n_0$, p(k) implies p(k+1). I.e, if p(k) is true, then p(k+1) is true also



From Ralph Grimaldi's discrete math book.

Why does induction work?

- Next we focus on a formal proof of this, because:
  - Some people may not be convinced by the informal one
  - The proof itself illustrates some important proof techniques

Proof that induction works (Overview)

- Let p be a property (p: int → boolean).
- Hypothesis:
  a) $p(n_0)$ is true.
  b) For all $k \geq n_0$, p(k) implies p(k+1). I.e, if p(k) is true, then p(k+1) is true also
- Desired Conclusion: If n is any integer with $n \geq n_0$, then p(n) is true. **If we can prove this, then induction is a legitimate proof method.**
- Proof that the conclusion follows from the hypothesis:
- Let S be the set $\{n \geq n_0 : p(n) \text{ is false}\}$.
- It suffices to show that S is empty.
- We do it by contradiction.
  - Assume that S is non-empty and show that this leads to a contradiction.

Proof that induction works

- Hypothesis:
  a) $p(n_0)$ is true.
  b) For all $k \geq n_0$, p(k) implies p(k+1).

  Desired Conclusion: If n is any integer with $n \geq n_0$, then p(n) is true. **If this conclusion is true, induction is a legitimate proof method.**
- **Proof**: Assume a) and b). Let S be the set $\{n \geq n_0 : p(n) \text{ is false}\}$. **We want to show that S is empty**; we do it by contradiction.
  - **Assume that S is non-empty.** Then the well-ordering principle says that S has a smallest element (call it s_{min}). We try to show that this leads to a contradiction.
  - Note that $p(s_{min})$ has to be false. **Why?**
  - s_{min} cannot be n_0, by hypothesis (a). Thus s_{min} must be $> n_0$. **Why?**
  - Thus smin-1 $\geq n_0$. Since s_{min} is the smallest element of S, $s_{min}-1$ cannot be an element of S. **What does this say about p (s_{min} – 1)?**
    - $p(s_{min}-1)$ is true.
  - By hypothesis (b), using the k= $s_{min}-1$ case, $p(s_{min})$ is also true. This **contradicts** the previous statement that $p(s_{min})$ is false.
  - Thus **the assumption that led to this contradiction** (S is nonempty) **must be false.**
  - Therefore S is empty, and p(n) is true for all $n \geq n_0$.

Next time …

- We will use the Principle of Mathematical induction to prove some properties of integers.

Big-Oh, Big-Omega, Big-Theta

Review of the concepts

Asymptotic analysis

- We only really care what happens when N (the size of a problem) gets large.
- Is the function linear? quadratic? etc.

Figure 5.1
Running times for small inputs

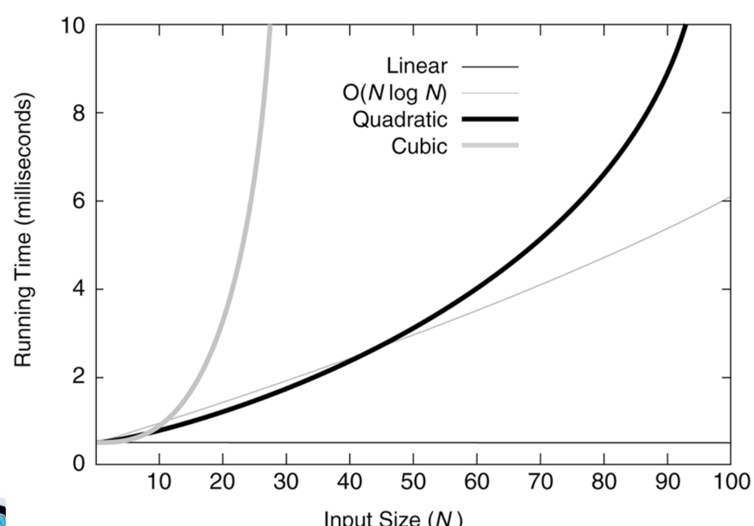


Figure 5.2

Running times for moderate inputs

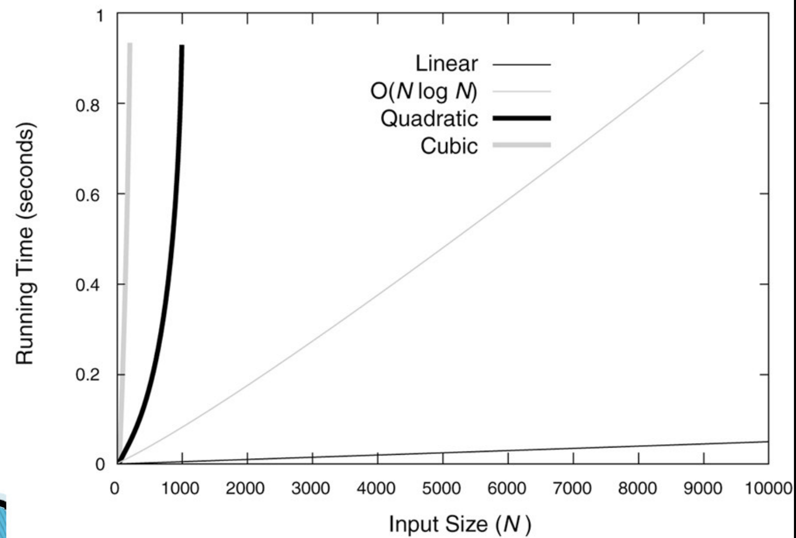


Figure 5.3

Functions in order of increasing growth rate

FUNCTION	NAME
c	Constant
$\log N$	Logarithmic
$\log^2 N$	Log-squared
N	Linear
$N \log N$	N log N ← a.k.a "log linear"
N^2	Quadratic
N^3	Cubic
2^N	Exponential

Data Structures & Problem Solving using JAVA/2E Mark Allen Weiss © 2002 Addison Wesley

Informal definitions

- f(N) is O(g(N)) means …
- f(N) is Ω(g(N)) means …
- f(N) is Θ(g(N)) means …
- Relationships between
  - f(x) = 3x – 2 and g(x) = x
  - f(x) = x and g(x) = x^2.

Toward a formal definition of big-oh

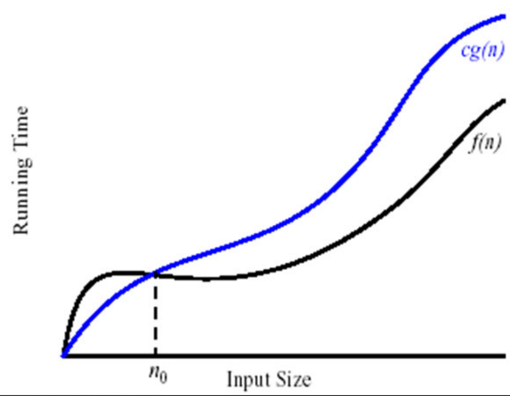
The definition has a lot in common with a particular limit definition.

Formal definition of $\lim_{x \to \infty} f(x) = a$

Formal definition of f(N) is O(g(N)) is similar:

And so is the definition of f(N) is Ω(g(N))

- The "Big-Oh" Notation
  - given functions f(n) and g(n), we say that f(n) is ***O***(g(n)) if and only if f(n) ≤ c g(n) for n n_0 ≥
  - c and n_0 are constants, f(n) and g(n) are functions over non-negative integers



Big Oh examples

- For this discussion, assume that all functions have non-negative values, and that we only care about x≥0.
  For any function g(x), O(g(x)) is a set of functions.
  We say that a function **f(x) is (in) O(g(x))** if there exist two positive constants **c** and **x_0** such that for all $x \geq x_0$, $f(x) \leq c\, g(x)$. **Rewrite using ∀ and ∃ notation**
- **All that we must do to prove that f(x) is O(g(x)) is produce a pair of numbers c and x0 that work for that case.**
- f(x) = x, g(x) = x^2.
- f(x) = x, g(x) = 3x.
- f(x) = x + 12, g(x) = x.
  We can choose c =3 and x_0 = 6, or c = 4 and x_0 = 4.
- f(x) = x + sin(x)
- f(x) = x^2 + sqrt(x)

Answers to examples

- For this discussion, assume that all functions have non-negative values, and that we only care about $x \geq 0$. For any function g(x), O(g(x)) is a set of functions We say that a function f(x) is (in) O(g(x)) if there exist two positive constants **c** and $\mathbf{x_0}$ such that for all $x \geq x_0$, $f(x) \leq c\, g(x)$. **Rewrite using ∀ and ∃ notation**
- **So all we must do to prove that f(x) is O(g(x) is produce two such constants.**
- f(x) = x + 12, g(x) = ???.
  - g(x) = x. Then c =3 and x_0 = 6, or c = 4 and x_0 = 4, etc.
- f(x) = x + sin(x): g(x) = x, c = 2, x_0 = 1
- f(x) = x^2 + sqrt(x): g(x) = x^2, c=2, x0 = 1

- Simple Rule: Drop lower order terms and constant factors.
  - $7n - 3$ is $O(n)$
  - $8n^2 \log n + 5n^2 + n$ is $O(n^2 \log n)$

- Special classes of algorithms:
  - logarithmic: $O(\log n)$
  - linear $O(n)$
  - quadratic $O(n^2)$
  - polynomial $O(n^k)$, $k \geq 1$
  - exponential $O(a^n)$, $n > 1$

- "Relatives" of the Big-Oh
  - $\Omega(f(n))$: Big Omega
  - $\Theta(f(n))$: Big Theta

Limits and asymptotics

- Consider the limit $$\lim_{n\to\infty} \frac{f(n)}{g(n)}$$
- What does it say about asymptotics if this limit is zero, nonzero, infinite?
- We could say that knowing the limit is a sufficient but not necessary condition for recognizing big-oh relationships.
- It will be sufficient for most examples in this course.
- Challenge: Use the formal definition of limit and the formal definition of big-oh to prove these properties.

Apply this limit property to the following pairs of functions

1. N and N^2
2. $N^2 + 3N + 2$ and N^2
3. N + sin(N) and N
4. log N and N
5. N log N and N^2
6. N^a and N^n
7. a^N and b^N (a < b)
8. $\log_a N$ and $\log_b N$ (a < b)
9. N! and N^N

Big-Oh Style

- **Give tightest bound you can**
  - Saying that **3N+2 is $O(N^3)$** is true, but not as useful as saying it's O(N) [What about **$\Theta(N^3)$** ?]
- **Simplify:**
  - You *could* say:
  - 3n+2 is O(5n-3log(n) + 17)
  - and it would be technically correct…
  - But **3n+2 is O(n)** is better
- **But… if I ask "true or false: 3n+2 is $O(n^3)$", what's the answer?**
  - True!
  - There may be "trick" questions like this on assignments and exams.
  - But they aren't really tricks, just following the big-Oh definition!