Analysis of Algorithms
CS 477/677

Instructor: Monica Nicolescu

Lecture 2

History Bit

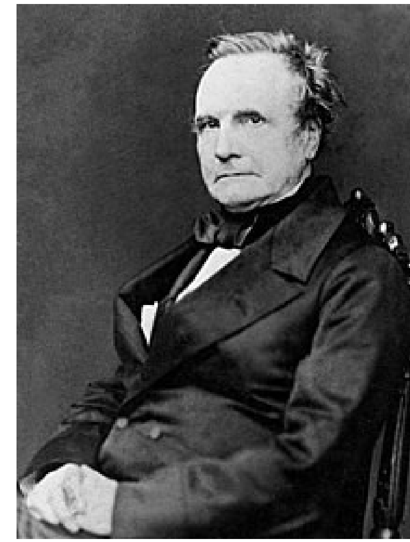
- First mechanical computers: difference engine, analytical engine
  - Arithmetic logic unit, control flow (branching, loops) and memory
  - 4 arithmetic operations, comparisons, square roots

On two occasions I have been asked [*by members of Parliament*], *'Pray, Mr. Babbage, if you put into the machine wrong figures, will the right answers come out?' I am not able rightly to apprehend the kind of confusion of ideas that could provoke such a question.*

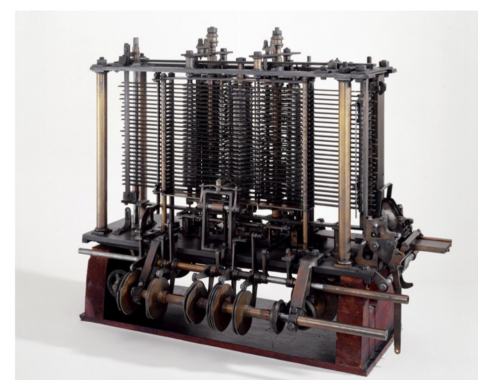
The whole of the developments and operations of analysis are now capable of being executed by machinery. ... As soon as an Analytical Engine exists, it will necessarily guide the future course of science. Whenever any result is sought by its aid, the question will then arise—By what course of calculation can these results be arrived at by the machine in the shortest time?

***Passages from the Life of a Philosopher* (London 1864)**

Charles Babbage
1719-1871



The analytical engine



Algorithm Analysis

- The amount of resources used by the algorithm
  - Space
  - Computational time
- Running time:
  - The number of primitive operations (steps) executed before termination
- Order of growth
  - The leading term of a formula
  - Expresses the behavior of a function toward infinity

Asymptotic Notations

- A way to describe behavior of functions in the limit
  - How we indicate running times of algorithms
  - Describe the running time of an algorithm as n grows to ∞
- O notation: asymptotic "less than": $f(n)$ "$\leq$" $g(n)$
- Ω notation: asymptotic "greater than": $f(n)$ "$\geq$" $g(n)$
- Θ notation: asymptotic "equality": $f(n)$ "$=$" $g(n)$

Logarithms

- In algorithm analysis we often use the notation "**log n**" without specifying the base

Binary logarithm $lgn = log_2 n$

$lnn = log_e n$

$$log^k n = (logn)^k$$

$$loglogn = log(logn)$$

$$log\, x^y = \text{¿} ylogx$$

$$logxy = \text{¿} logx + logy$$

$$log \frac{x}{y} = \text{¿} logx - logy$$

$$log_a x = \text{¿} log_a b \times log_b x$$

$$a^{log_b x} = x^{log_b a}$$

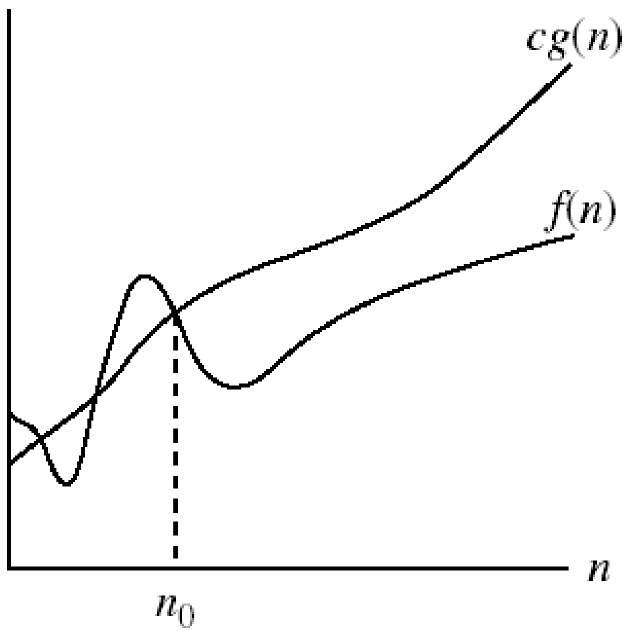
Asymptotic Notations - Examples

- For each of the following pairs of functions, either $f(n)$ is $O(g(n))$, $f(n)$ is $\Omega(g(n))$, or $f(n)$ is $\Theta(g(n))$. Determine which relationship is correct.
  - $f(n) = \log n^2$; $g(n) = \log n + 5$ $\quad f(n) = \Theta(g(n))$
  - $f(n) = n$; $g(n) = \log n^2$ $\quad f(n) = \Omega(g(n))$
  - $f(n) = \log \log n$; $g(n) = \log n$ $\quad f(n) = O(g(n))$
  - $f(n) = n$; $g(n) = \log^2 n$ $\quad f(n) = \Omega(g(n))$
  - $f(n) = n \log n + n$; $g(n) = \log n$ $\quad f(n) = \Omega(g(n))$
  - $f(n) = 10$; $g(n) = \log 10$ $\quad f(n) = \Theta(g(n))$
  - $f(n) = 2^n$; $g(n) = 10n^2$ $\quad f(n) = \Omega(g(n))$
  - $f(n) = 2^n$; $g(n) = 3^n$ $\quad f(n) = O(g(n))$

Asymptotic notations

- *O-notation*

$$O(g(n)) = \{f(n) : \text{there exist positive constants } c \text{ and } n_0 \text{ such that } 0 \le f(n) \le cg(n) \text{ for all } n \ge n_0\}.$$



- Intuitively: O(g(n)) = the set of functions with a smaller or same order of growth as g(n)

$g(n)$ is an ***asymptotic upper bound*** for $f(n)$.

Examples

$O(g(n)) = \{f(n) :$ there exist positive constants c and n_0 such that $0 \le f(n) \le cg(n)$ for all $n \ge n_0\}$.

- $2n^2 = O(n^3)$: $\exists\ c, n_0$ such that $2n^2 \le cn^3 \Rightarrow 2 \le cn$
  $\Rightarrow c = 1$ and $n_0 = 2$
- $n^2 = O(n^2)$: $\exists\ c, n_0$ such that $n^2 \le cn^2 \Rightarrow c \ge 1$
  $\Rightarrow c = 1$ and $n_0 = 1$
- $1000n^2+1000n = O(n^2)$:

  $\exists\ c, n_0$ such that $1000n^2+1000n \le cn^2$

  $1000n^2+1000n \le 1000n^2+ 1000n^2 = 2000n^2$

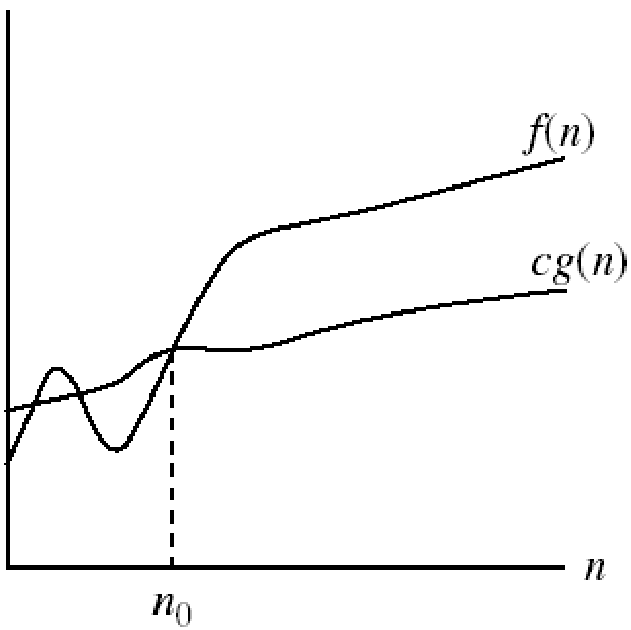
  $\Rightarrow c=2000$ and $n_0 = 1$

  Fact: $1000n \le 1000n^2$ $\forall$ (for all) $n \ge 1$
- $n = O(n^2)$: $\exists\ c, n_0$ such that $n \le cn^2 \Rightarrow cn \ge 1$
  $\Rightarrow c = 1$ and $n_0 = 1$

Asymptotic notations (cont.)

- **Ω** - *notation*

$$\Omega(g(n)) = \{f(n) : \text{there exist positive constants } c \text{ and } n_0 \text{ such that } 0 \le cg(n) \le f(n) \text{ for all } n \ge n_0\}\ .$$



- Intuitively: **Ω**(g(n)) = the set of functions with a larger or same order of growth as g(n)

$g(n)$ is an ***asymptotic lower bound*** for $f(n)$.

Examples

$\Omega(g(n)) = \{f(n) :$ there exist positive constants c and n_0 such that $0 \le cg(n) \le f(n)$ for all $n \ge n_0\}$.

- $5n^2 = \mathbf{\Omega}(n)$

$\exists\ c, n_0$ such that: $0 \le cn \le 5n^2 \Rightarrow cn \le 5n^2 \Rightarrow c = 1$ and $n_0 = 1$

- $100n + 5 \neq \mathbf{\Omega}(n^2)$

use proof by contradiction: *assume statement True*

$\exists\ c, n_0$ such that: $0 \le cn^2 \le 100n + 5$

$100n + 5 \le 100n + 5n\ (\forall\ n \ge 1) = 105n$

$cn^2 \le 105n \Rightarrow n(cn - 105) \le 0$

Since n is positive $\Rightarrow cn - 105 \le 0 \Rightarrow n \le 105/c$

$\Rightarrow$ contradiction: n cannot be smaller than a constant

Examples

$\Omega(g(n)) = \{f(n) :$ there exist positive constants c and n_0 such that $0 \le cg(n) \le f(n)$ for all $n \ge n_0\}$.

- $50n^2 + n = \mathbf{\Omega}(n)$

  $\exists$ c, n_0 such that: $0 \le cn \le 50n^2 + n$

  $50n^2 + n \ge 50n^2 \ge 50n$

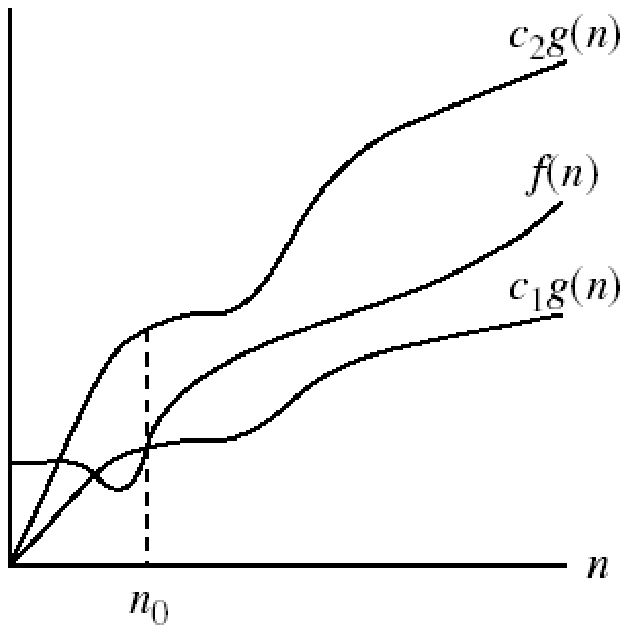
  Could use $c = 50$ and $n_0 = 1$

- $n = \mathbf{\Omega}(2n)$, $n^3 = \mathbf{\Omega}(n^2)$, $n = \mathbf{\Omega}(\log n)$

Asymptotic notations (cont.)

- *Θ-notation*

$$\Theta(g(n)) = \{ f(n) : \text{ there exist positive constants } c_1, c_2, \text{ and } n_0 \text{ such that } 0 \le c_1 g(n) \le f(n) \le c_2 g(n) \text{ for all } n \ge n_0 \} .$$



- Intuitively Θ(g(n)) = the set of functions with the same order of growth as g(n)

$g(n)$ is an ***asymptotically tight bound*** for $f(n)$.

Examples

$\Theta(g(n)) = \{f(n) :$ there exist positive constants c_1, c_2, and n_0 such that $0 \le c_1 g(n) \le f(n) \le c_2 g(n)$ for all $n \ge n_0\}$.

- $= \Theta(n^2)$ $\exists\ c_1, c_2, n_0$ such that
  - $\forall n \ge 0 \Rightarrow c_2 = 1/2$
  - $(\forall n \ge 2) = \Rightarrow c_1 = 1/4$

Subtract a larger quantity

- $n \ne \Theta(n^2)$: $\exists\ c, n_0$ such that $c_1 n^2 \le n \le c_2 n^2$

$\Rightarrow$ only holds for: $n \le 1/c_1$

Examples

- $n \neq \Theta(n^2)$: $\exists\ c, n_0$ such that $c_1 n^2 \leq n \leq c_2 n^2$

  $\Rightarrow$ only holds for: $n \leq 1/c_1$

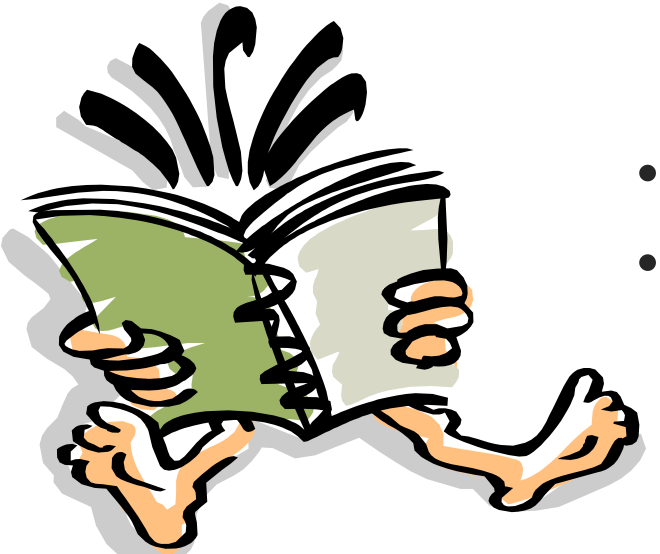
- $6n^3 \neq \Theta(n^2)$: $\exists\ c, n_0$ such that $c_1 n^2 \leq 6n^3 \leq c_2 n^2$

  $\Rightarrow$ only holds for: $n \leq c_2 /6$

- $n \neq \Theta(\log n)$: $\exists\ c, n_0$ such that $c_1 \log n \leq n \leq c_2 \log n$

  $\Rightarrow c_2 \geq n/\log n,\ \forall\ n \geq n_0$ – impossible

Readings



- Chapter 3
- Apendix A