Big Oh and Unit Testing

CSSE 221

Fundamentals of Software Development Honors

Rose-Hulman Institute of Technology

ROSE-HULMAN
INSTITUTE OF TECHNOLOGY
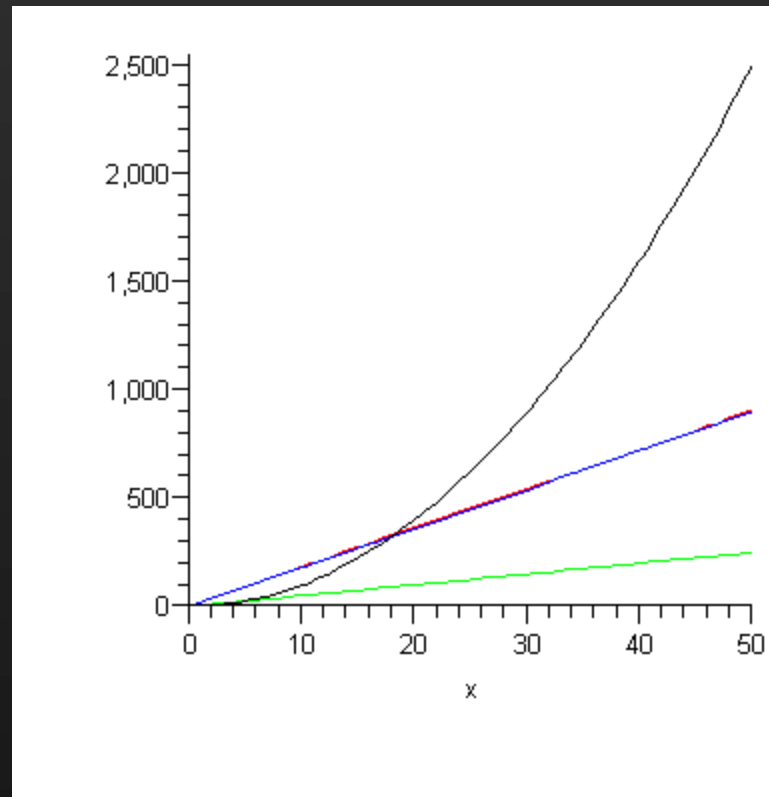
Efficiency is important

- Example?
- Not all a software problem
- Algorithms
  - Inherent complexity
  - Assume time spent is a function of the size of the input
  - Big-Oh focuses on the **most important** part of the function!
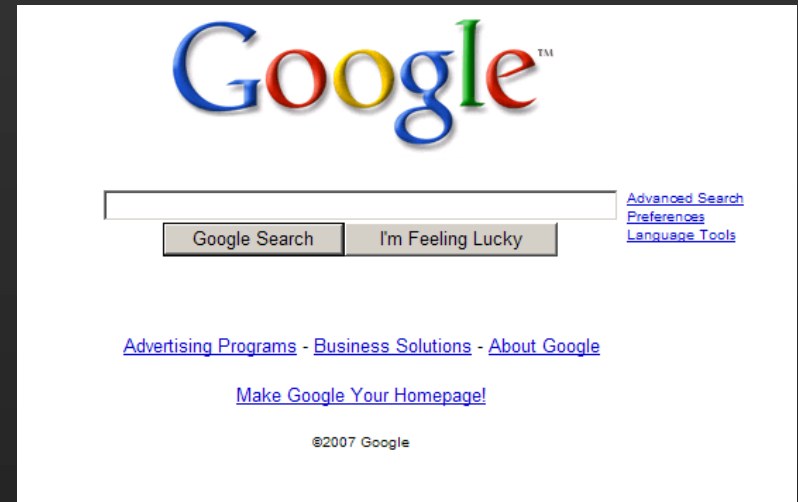
Now: plot y=18x + 5, y=18x, y = 5x, y=x^2 Which grows most quickly?

Efficiency is important

$plot([18x + 5, \ 18x, \ 5x, \ x^2]$
$, \ x = 0..50, \ color = [red, blue, green, black])$

y=x²

y=18x + 5
y=18x

y = 5x

- **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 1
  - exponential $O(a^n)$, $n > 1$
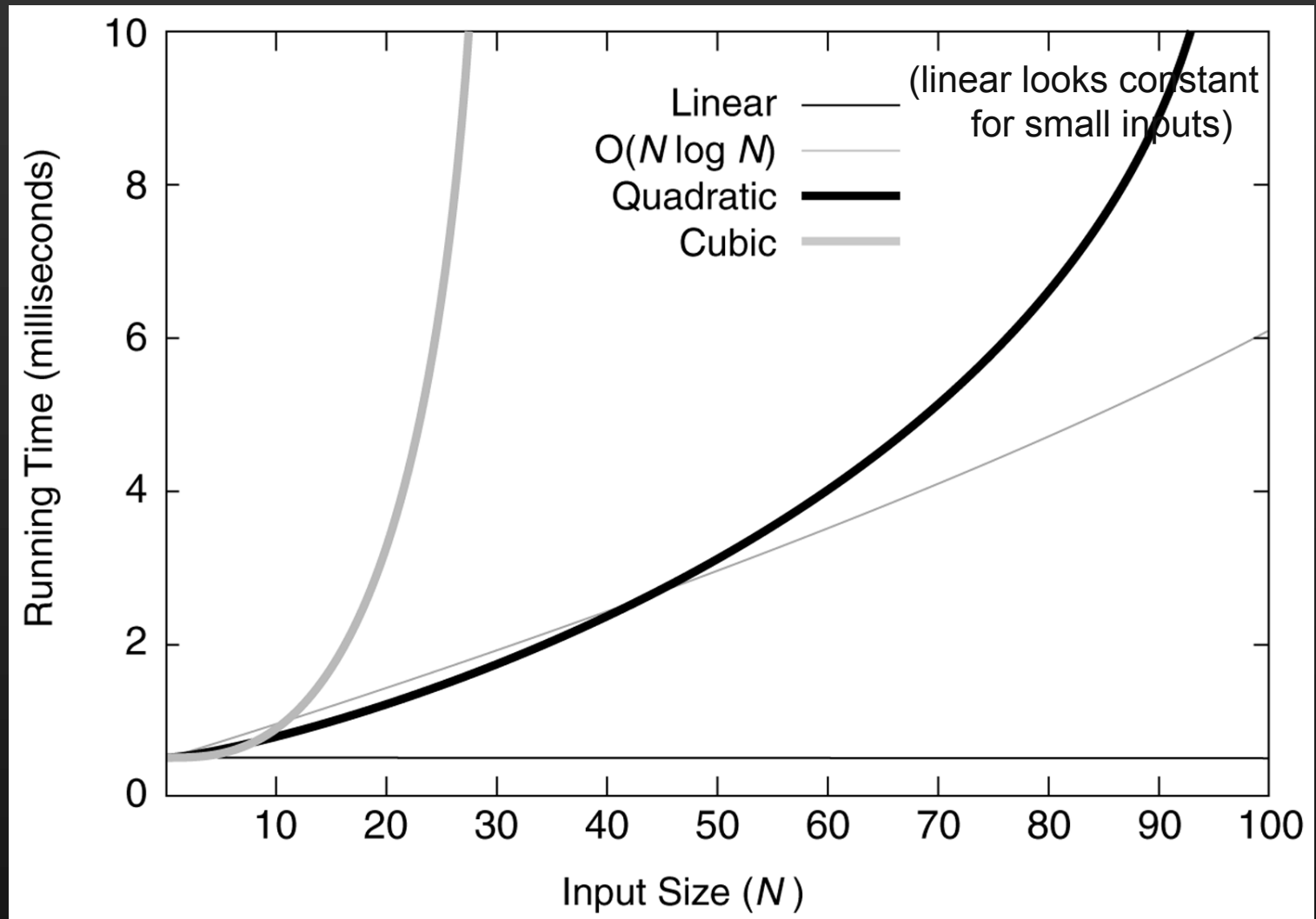
Also: **constant: O(1)**

Figure 5.1
Running times for small inputs

Figure 5.2
Running times for moderate inputs