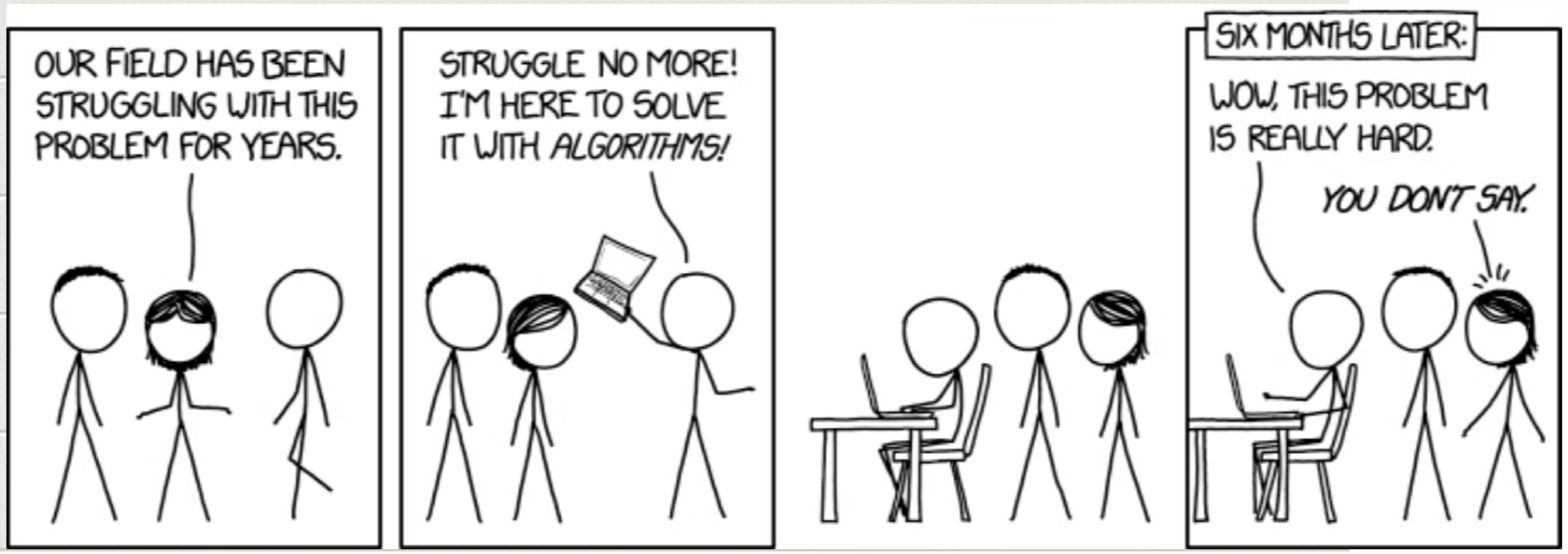
CSCI 3100: Algorithms

<u>Today</u>: - Syllabus
- Intro to algorithms



algorithm
noun

Word used by programmers when they do not want to explain what they did.

Question: What <u>is</u> an algorithm?

(Side question: What is a program?)

Origins

- Not Greek "algos" : ?
- 9th century writer & mathematician
  Abu Abd Allāh Muhammad ibn Mūsā
  Al-Khwārizmī
  (also where "algebra" came from, & invented 0 as a place holder)

  - Later known as "algoism", popularized by Fibonacci

The usual silly examples

```
BOTTLESOFBEER(n):
  For i ← n down to 1
      Sing "i bottles of beer on the wall, i bottles of beer,"
      Sing "Take one down, pass it around, i − 1 bottles of beer on the wall."
  Sing "No bottles of beer on the wall, no bottles of beer,"
  Sing "Go to the store, buy some more, n bottles of beer on the wall."
```

f Arabic numerals; it was still taught in elementary schools in Ea
entury. This algorithm was also commonly used by early digital co
: integer multiplication directly in hardware.

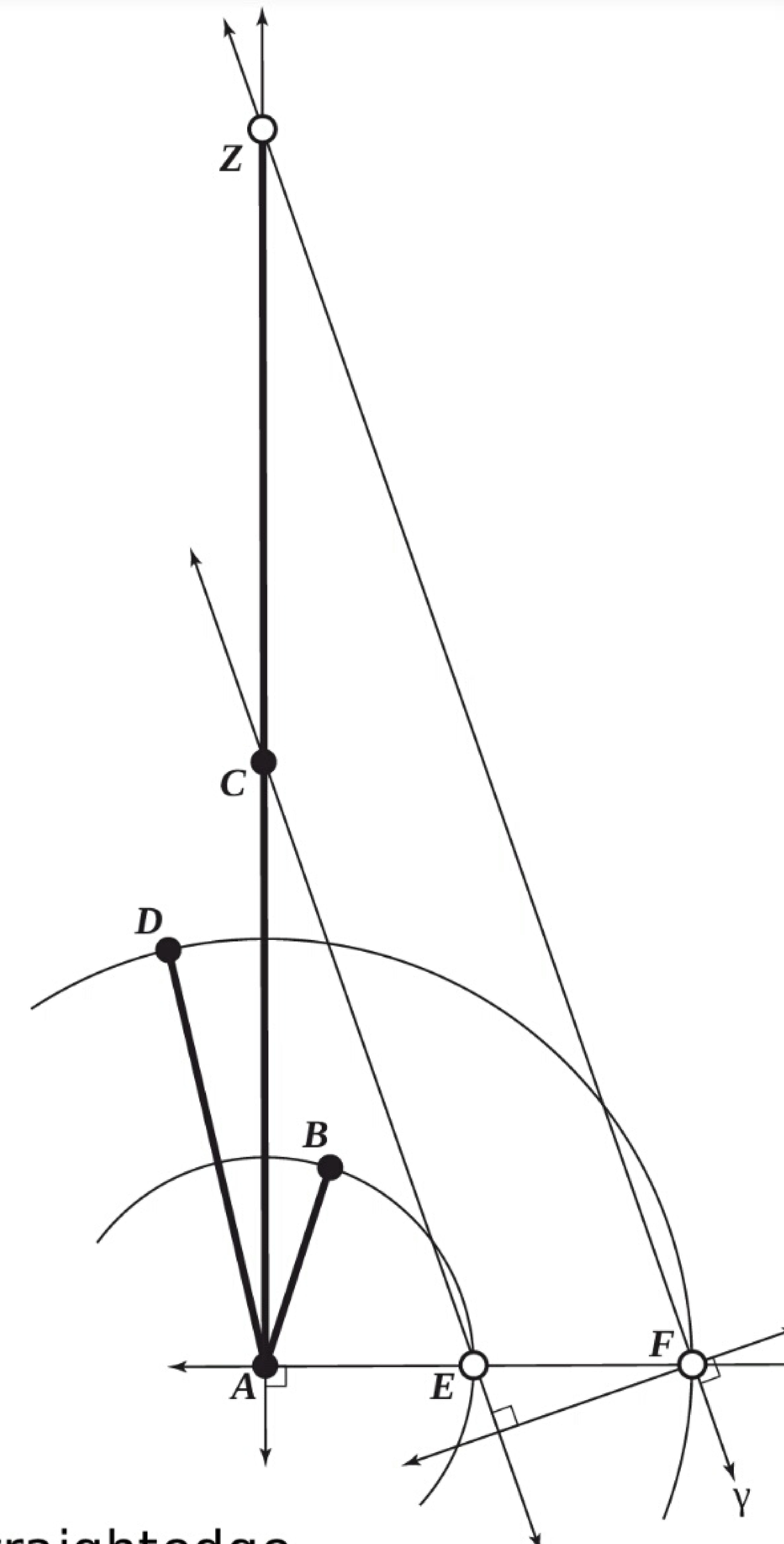
```
PEASANTMULTIPLY(x, y):
  prod ← 0
  while x > 0
      if x is odd
          prod ← prod + y
      x ← ⌊x/2⌋
      y ← y + y
  return p
```

x	y		$prod$
			0
123	$+\,456$	$=$	456
61	$+\,912$	$=$	1368
30	~~1824~~		
15	$+\,3648$	$=$	5016
7	$+\,7296$	$=$	12312
3	$+\,14592$	$=$	26904
1	$+\,29184$	$=$	**56088**

ıt multiplication algorithm breaks the difficult task of general mu

$\langle\langle$*Construct the line perpendicular to ℓ and passing through P.*$\rangle\rangle$
<u>RightAngle(ℓ, P):</u>
  Choose a point $A \in \ell$
  $A, B \leftarrow$ Intersect(Circle(P, A), ℓ)
  $C, D \leftarrow$ Intersect(Circle(A, B), Circle(B, A))
  return Line(C, D)

$\langle\langle$*Construct a point Z such that $|AZ| = |AC||AD|/|AB|$.*$\rangle\rangle$
<u>MultiplyOrDivide(A, B, C, D):</u>
  $\alpha \leftarrow$ RightAngle(Line(A, C), A)
  $E \leftarrow$ Intersect(Circle(A, B), α)
  $F \leftarrow$ Intersect(Circle(A, D), α)
  $\beta \leftarrow$ RightAngle(Line(E, C), F)
  $\gamma \leftarrow$ RightAngle(β, F)
  return Intersect(γ, Line(A, C))



Multiplying or dividing using a compass and straightedge.

[4]In fact, some medieval English sources claim the Greek prefix “algo-” meant “art” or “introductio

Now a bad example:

BECOMEAMILLIONAIREANDNEVERPAYTAXES:

Get a million dollars.

If the tax man comes to the door and says, "*You have never paid taxes!*"

Say "*I forgot.*"

Why is this bad?

Some tips for this class:

- Don't write actual code!
  - essentially, goal is to write the comments that should be in the program
- But - don't write English prose either!
  - use loops + data structures
- Indent and avoid brackets

More tips:

- Meaningful variable names, please!
- Keep a statement on a single line
- Find a good balance between words and math:

  Insert x into A

  INSERT(x, A)

  $X \leftarrow X \cup \{a\}$

3 parts to every algorithm:

1

2

3

+ sometimes 4:

This week: why you should have paid attention in discrete math & data structures!

Topics to recall:

Runtimes:

What is big-O analysis?

Why use it?

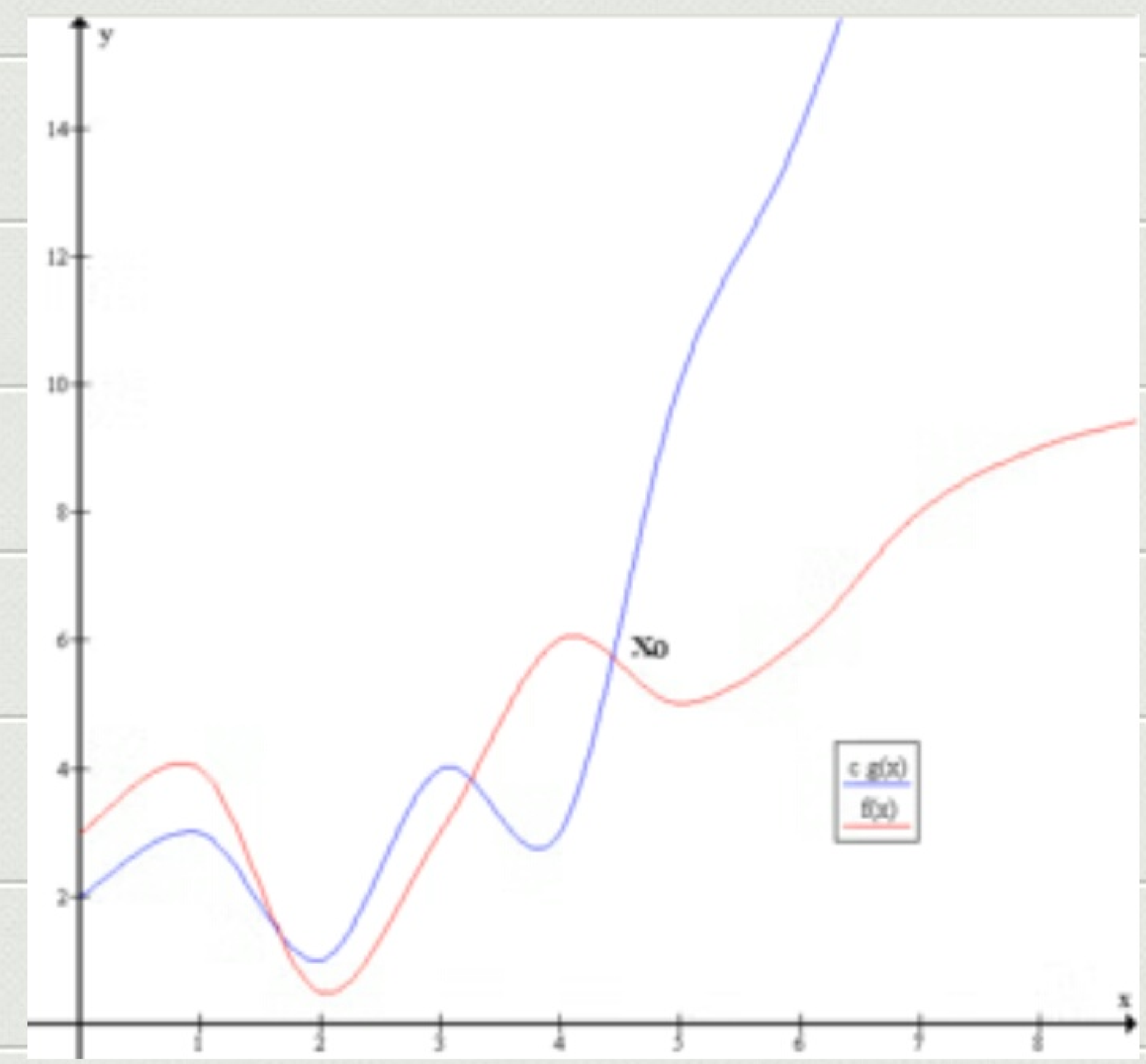
Formal dfn:

Let f & g be functions $\mathbb{R} \to \mathbb{R}$ (or $\mathbb{Z} \to \mathbb{R}$). We say that:

$$f(n) = O(g(n))$$

if $\exists$ constants c & n_0 s.t.

$$|f(n)| \le c|g(n)|$$
$$\forall n > n_0$$

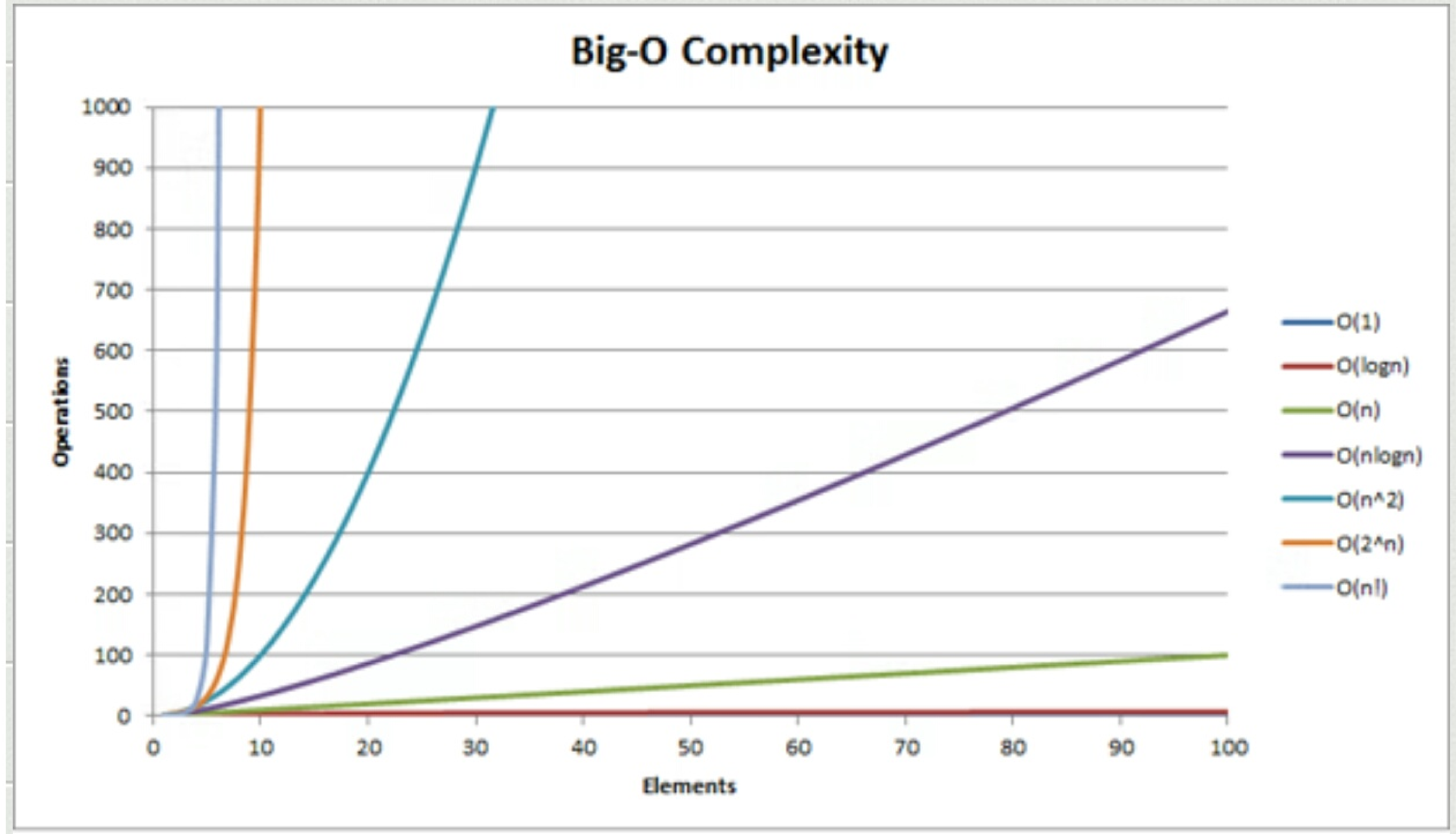


Big-O: functions ranking

BETTER

- $O(1)$ constant time
- $O(\log n)$ log time
- $O(n)$ linear time
- $O(n \log n)$ log linear time
- $O(n^2)$ quadratic time
- $O(n^3)$ cubic time
- $O(2^n)$ exponential time

WORSE



Example proof:

$$f(x) = x^2 + 2x + 1 \quad \text{is} \quad O(x^2)$$

pf:

Key thm:

Let $f(x)$ be a polynomial of degree n,
so $f(x) = \sum_{i=0}^{n} a_i x^i$

where each $a_i \in \mathbb{R}$.

Then $f(x) = O(x^n)$.

pf sketch:

Induction: recursion's twin

A method of proving a statement which depends on the statement being true for smaller values

Required pieces:

Aside: I think of this as "automating" a proof:

Show true for $n=1$.

Show if n holds, then $n+1$ must also.

$\Rightarrow$ Get all n for free!

Example: $\sum_{i=0}^{n} i =$

Next time:

- Recursion as an algorithmic technique
- Even more induction