COT 5407: Introduction to Algorithms

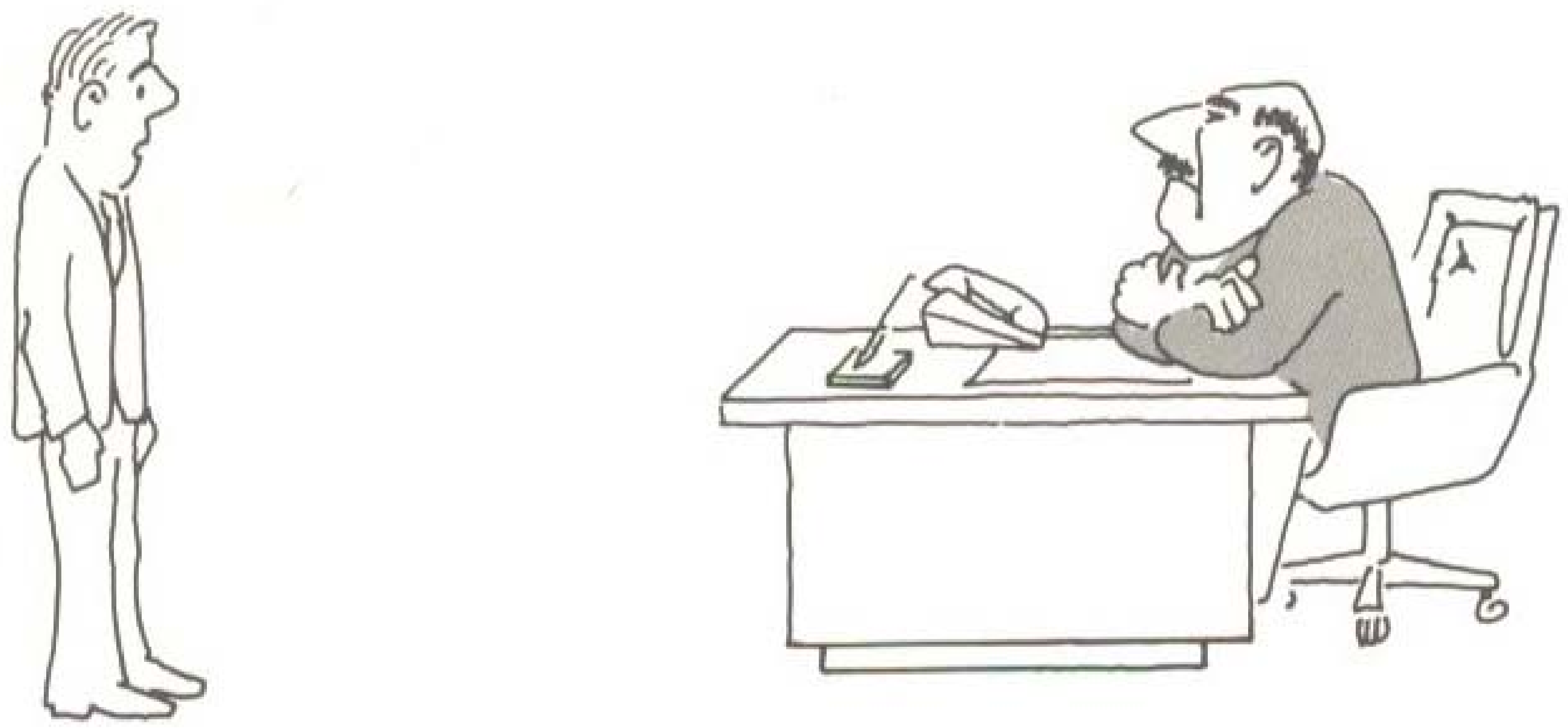
Giri Narasimhan

ECS 254; Phone: x3748

giri@cis.fiu.edu

www.cis.fiu.edu/~giri/teach/5407F07.html

Why should I care about Algorithms?



"I can't find an efficient algorithm, I guess I'm just too dumb."

Cartoon from *Intractability* by Garey and Johnson

More questions you should ask

- Who should know about Algorithms?
- Is there a future in this field?
- Would I ever need it if I want to be a software engineer or work with databases?

Why are theoretical results useful?

“I can’t find an efficient algorithm, because no such algorithm is possible!”

Cartoon from *Intractability* by Garey and Johnson

Why are theoretical results useful?

"I can't find an efficient algorithm, but neither can all these famous people."

Cartoon from *Intractability* by Garey and Johnson

Evaluation

• Exams (2)	50%
• Homework Assignments	35%
• Semester Project	10%
• Class Participation	5%

History of Algorithms

The great thinkers of our field:

- **Euclid, 300 BC**
- **Bhaskara, 6th century**
- **Al Khwarizmi, 9th century**
- **Fibonacci, 13th century**
- **Babbage, 19th century**
- **Turing, 20th century**
- **von Neumann, Knuth, Karp, Tarjan, …**

Search

- You are asked to guess a number X that is known to be an integer lying in the range A through B. How many guesses do you need in the worst case?
  - Use **binary search**; Number of guesses = $\log_2(B-A)$
- You are asked to guess a positive integer X. How many guesses do you need in the worst case?
  - NOTE: No upper bound is known for the number.
  - Algorithm:
    - figure out B (by using **Doubling Search**)
    - perform binary search in the range B/2 through B.
  - Number of guesses = $\log_2 B + \log_2(B - B/2)$
  - Since X is between B/2 and B, we have: $\log_2(B/2) < \log_2 X$,
  - Number of guesses < $2\log_2 X - 1$

Polynomials

- Given a polynomial
  - $p(x) = a_0 + a_1 x + a_2 x^2 + \dots + a_{n-1} x^{n-1} + a_n x^n$

  compute the value of the polynomial for a given value of x.
- How many additions and multiplications are needed?
  - Simple solution:
    - Number of additions = n
    - Number of multiplications = 1 + 2 + … + n = n(n+1)/2
  - Improved solution using **Horner's rule**:
    - $p(x) = a_0 + x(a_1 + x(a_2 + \dots x(a_{n-1} + x\, a_n))\dots))$
    - Number of additions = n
    - Number of multiplications = n