Analyzing Algorithms

- Consider three solutions to SortByFreqs, also code used in Anagram assignment
  - Sort, then scan looking for changes
  - Insert into Set, then count each unique string
  - Find unique elements without sorting, sort these, then count each unique string

- We want to discuss trade-offs of these solutions
  - Ease to develop, debug, verify
  - Runtime efficiency
  - Vocabulary for discussion

Cost

> *"An engineer is someone who can do for a dime what any fool can do for a dollar."*

- Types of costs:
  - Operational
  - Development
  - Failure
- Is this program fast enough? What's your purpose? What's your input data?
- How will it *scale*?
- Measuring cost
  - Wall-clock or execution time
  - Number of times certain statements are executed
  - Symbolic execution times
    - Formula for execution time in terms of *input size*
  - Advantages and disadvantages?

Some complexity notes courtesy of Paul Hilfinger

Data processing example

- Scan a large (~ 10^7 bytes) file
- Print the 20 most frequently used words together with counts of how often they occur
- Need more specification?

- How do you do it?