ACS 560
Software Engineering

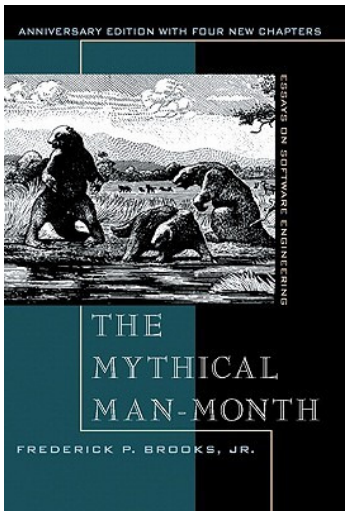
Fall 2013

Instructor: Dr. Zesheng Chen

Outline

- Why is programming fun?
- Why is programming hard?
- What is the proper metaphor for software development?
- What are methods to effectively and efficiently find the answer to a question during software development?
- What is the first principle of programming?

Why is programming fun?



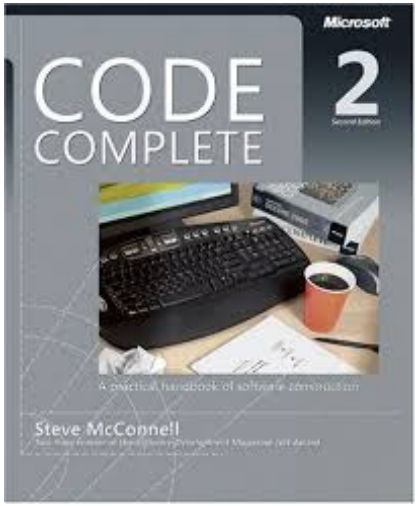
Why is programming fun?

- The sheer joy of making things
- The pleasure of making things that are useful to other people
- The feeling of achievement watching a complex puzzle-like objects to work in order
- The joy of always learning
- The delight of working in such a tractable medium

Why is programming hard?

- Complexity of software development
- Limitation of programmer (imperfectness)
- Communication between people
- Resource constrains (time/budget/etc.)
- Changing world (changing requirements/ changing technologies)

What is the proper metaphor for software development?



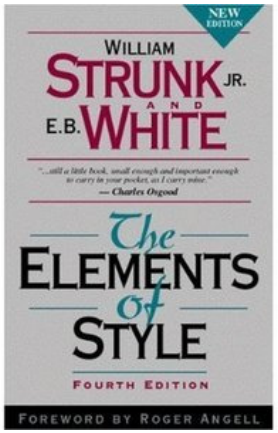
We use metaphors all the time

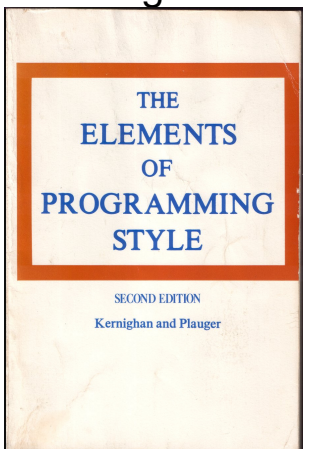
Why Metaphors for Software Development ?

- A software metaphor is more like a **searchlight** than a road map.
- Does not tell you where to find the answer, it tells you **how** to look for it.
- Serve more as a **heuristic** than it does as an algorithm.
- Help you think about your programming activities and to help to imagine **better** way of doing things.

What is the proper metaphor for software development?

- Software penmanship: Writing Code



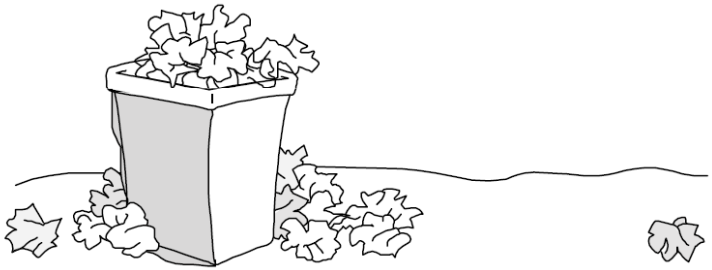


Many Similarities Between Writing Papers and Writing Code

- "A good writing is a bad writing re-written"
  - Refectoring
- Code review is similar to paper review
  - Peer code review

Objection to "Writing Code" Metaphors

- Lack of planning



F02xx01

Figure 2-1

The letter-writing metaphor suggests that the software process relies on expensive trial and error rather than careful planning and design.

What is the proper metaphor for software development?

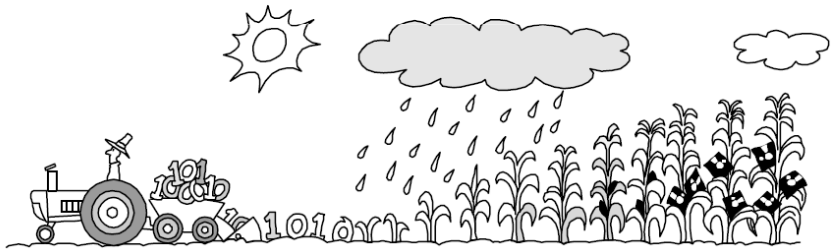
- Software farming: Growing a System
  - Design a piece
  - Code a piece
  - Test a piece
  - Add a piece to the system a little bit at a time
  - Similar to planting seeds and growing crops

Advocate to "Growing a System" Metaphor

- Incremental technique
- The idea has been used in Refactoring

Objection to "Growing a System" Metaphor

- You do not have any direct control over how the software develops.



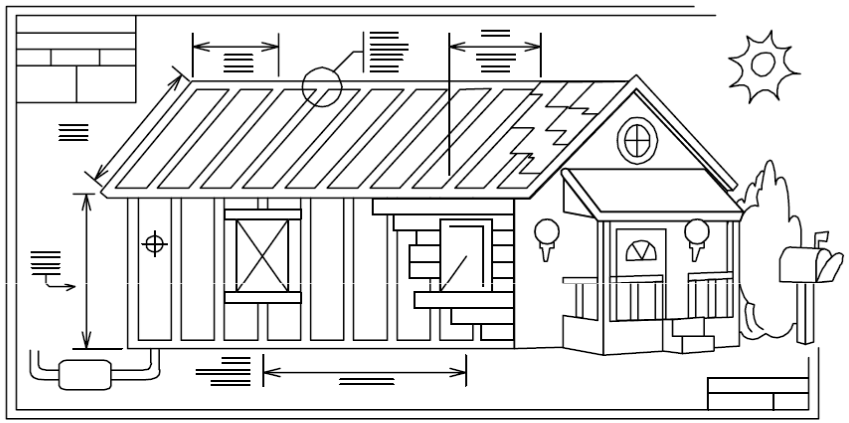
F02xx02

Figure 2-2

It's hard to extend the farming metaphor to software development appropriately.

What is the proper metaphor for software development?

- Software Construction: Building Software
  - Need careful planning
  - Need construction different modules together
  - Need verification



F02xx04

Figure 2-4

More complicated structures require more careful planning.

Objection to "Building Software" Metaphor

- When refactoring, it may be difficult to think about building a house

What are methods to effectively and efficiently find the answer to a question during software development?

Different Methods

- Google (or Bing or other search tools)
- Books
- People

Google should be your best friend

- Find the general information
  - How to find the information on the textbook ?
  - How to find how the textbook is used in other universities ?
- Look at:
  - http://www.google.com/advanced_search
  - http://www.sitepoint.com/10-tips-for-conducting-a-more-effective-google-search/

Google should be your best friend (Cont.)

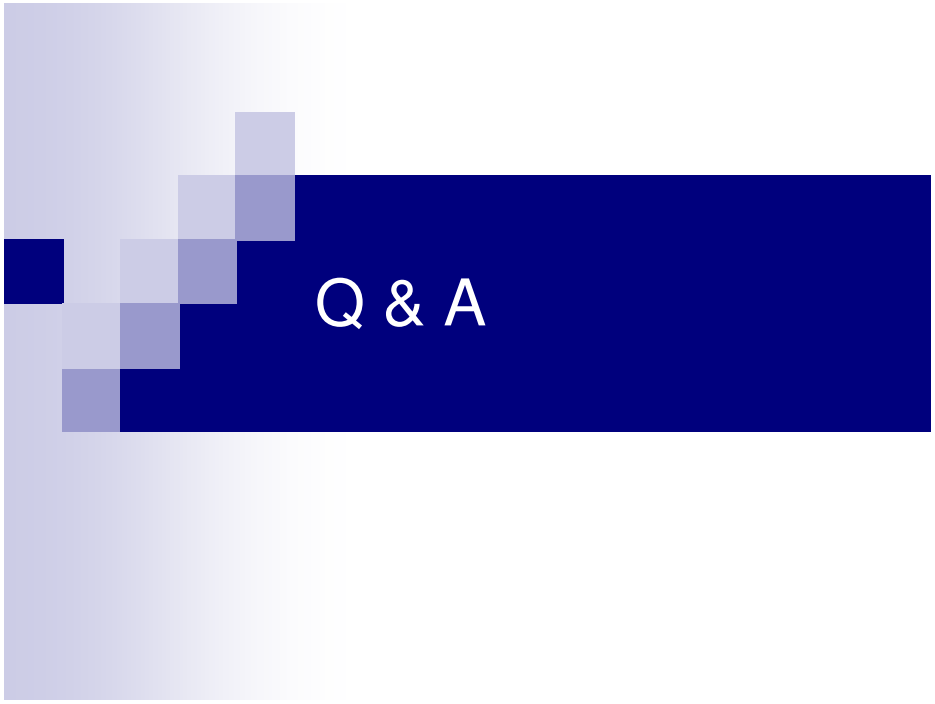
- Find the programming-related information
  - Function/method call (find the function call to strcmp() in C)
  - How to fix an issue in programming (google the error message)
  - How to learn a new language (DB/VB.Net)
- Work on HW 1 problems

What is the first principle of programming?

- DRY (**D**on't **R**epeat **Y**ourself)
  - Very useful and important
- KISS (**K**eep **I**t **S**imple **S**tupid)
- Understand the problem first.
- Write code like if it was you that would have to maintain that code.
- Be as lazy as possible.
- If it wasn't tested, it is broken.

Summary

- Five questions
  - Is high-level
  - Very important
  - Worth thinking
  - Will come back to some of them again and again
- Become an expert on Google searching


Q & A