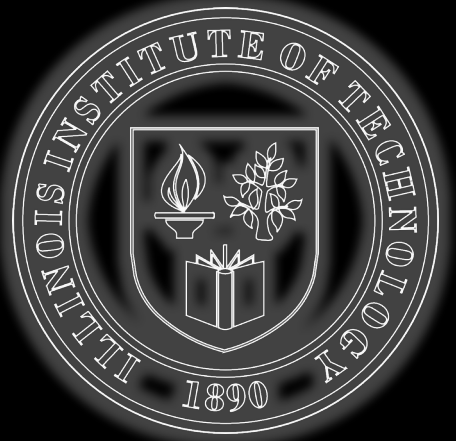
ILLINOIS TECH | College of Computing

ITMD-511 Chapter 4

Key Construction Decisions

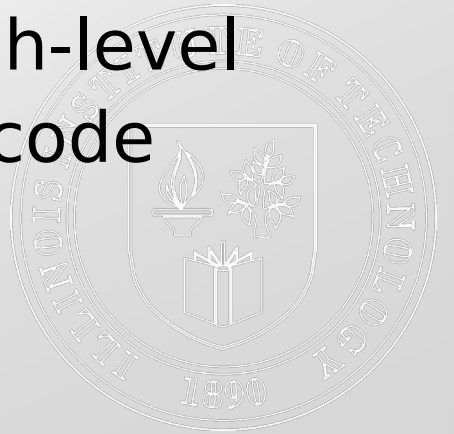


What is this chapter about?

- Once we have a groundwork laid out, we can turn attention towards more construction-specific decisions
- This focuses on items that individual programmers or technical leads are responsible for

Choice of Programming Languages

- Our book was written in the early 2000's, so even more languages are available for use now.
- C++, Java, Visual Basic, Smalltalk have been credited with improving productivity and comprehensibility.
  - Higher level languages are more expressive than lower-level ones
- The next slide has an image that just details how high-level statements in newer languages match up to C-level code

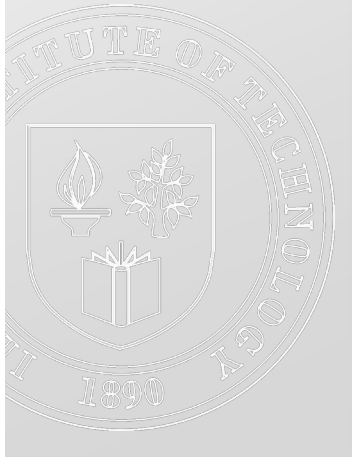


Choice of Programming Languages

Table 4-1 Ratio of High-Level-Language Statements to Equivalent C Code

Language	Level Relative to C
C	1
C++	2.5
Fortran 95	2
Java	2.5
Perl	6
Python	6
Smalltalk	6
Microsoft Visual Basic	4.5

Source: Adapted from *Estimating Software Costs* (Jones 1998), *Software Cost Estimation with Cocomo II* (Boehm 2000), and "An Empirical Comparison of Seven Programming Languages" (Prechelt 2000).

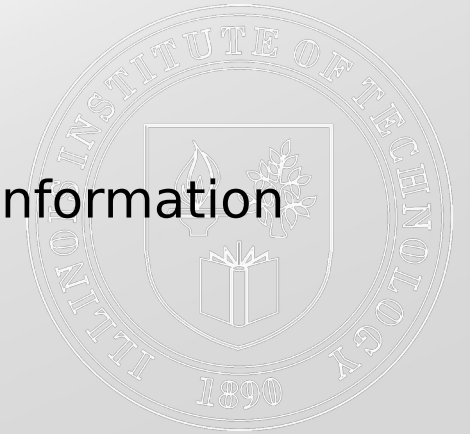


Choice of Programming Languages

- Some languages are better at expressing programming concepts over others
  - Natural parallel to English and Java / C++
    - Words in programming languages can help determine what thoughts or code we can express
  - Programmers learning C++ from Fortran disguised their C++ code as Fortran.
    - Relied on go-tos and global data rather than C++'s OOP concepts

Some Language Descriptions

- Many languages covered in the book
- C++
  - Provides classes, polymorphism, exception handling etc.
- Java
  - Object oriented language like C/C++
  - Converts Java source code to byte code, which is then run in the Java virtual machine
- PhP
  - Used in association with web pages to access and present database information
- Python
  - Object oriented language that runs in numerous environments



Programming Conventions

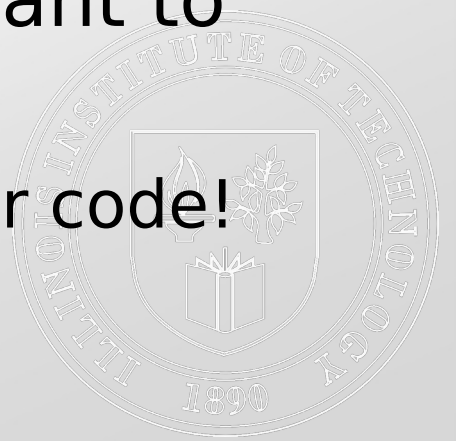
- In high-quality software, we see a relationship between the conceptual integrity of architecture and the low-level implementation
  - Variable names, class names, routine names, formatting and commenting conventions
- Make sure your code has a consistent style
  - Don’t try to paint a picture using 3 different styles, it will look more like a collage

Technology Waves

- Technology is constantly changing, and we need to be able to adapt and account for this
- Make this distinction
  - Programming **in** a language vs. programming **into** a language.
    - Programming **in** a language limits their thoughts to constructs that only operate in that language
    - Programming **into** a language allows thoughts that aren't constrained to the environment and allows us to express them using the language

Programming into a Language

- Important for our book, because we are learning **methodologies,** not a specific language!
- Coding principles are **not** specific to a language, they are able to be used across various areas.
- If a language lacks concepts or constructs you want to use, try to compensate for them.
  - Invent some conventions, libraries, and augment your code!



Major Construction Practices

- Some construction practices use pair programming or test-first development
- Others use solo development and formal inspections
- Feel free to explore the checklist that will be attached in the Week 2 module.

Key Points

- Every language has strengths and weaknesses. Be aware of these when using them
- Establish programming conventions before you begin programming, it is nearly impossible to change code to match them later
- There are a ton of construction practices, so feel free to choose which ones are best suited to your project

Key Points

- Ask yourself whether the programming practices you are using are a response to the programming language or are controlled by it.
  - Program **into** a language, don't program **in** a language
- Technology is always changing, so be adaptable and change expectations accordingly!