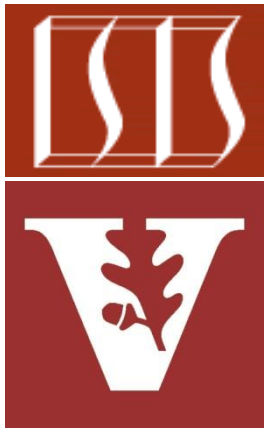
Overview of C++: Strategies for Learning C++

Douglas C. Schmidt

d.schmidt@vanderbilt.edu

www.dre.vanderbilt.edu/~schmidt



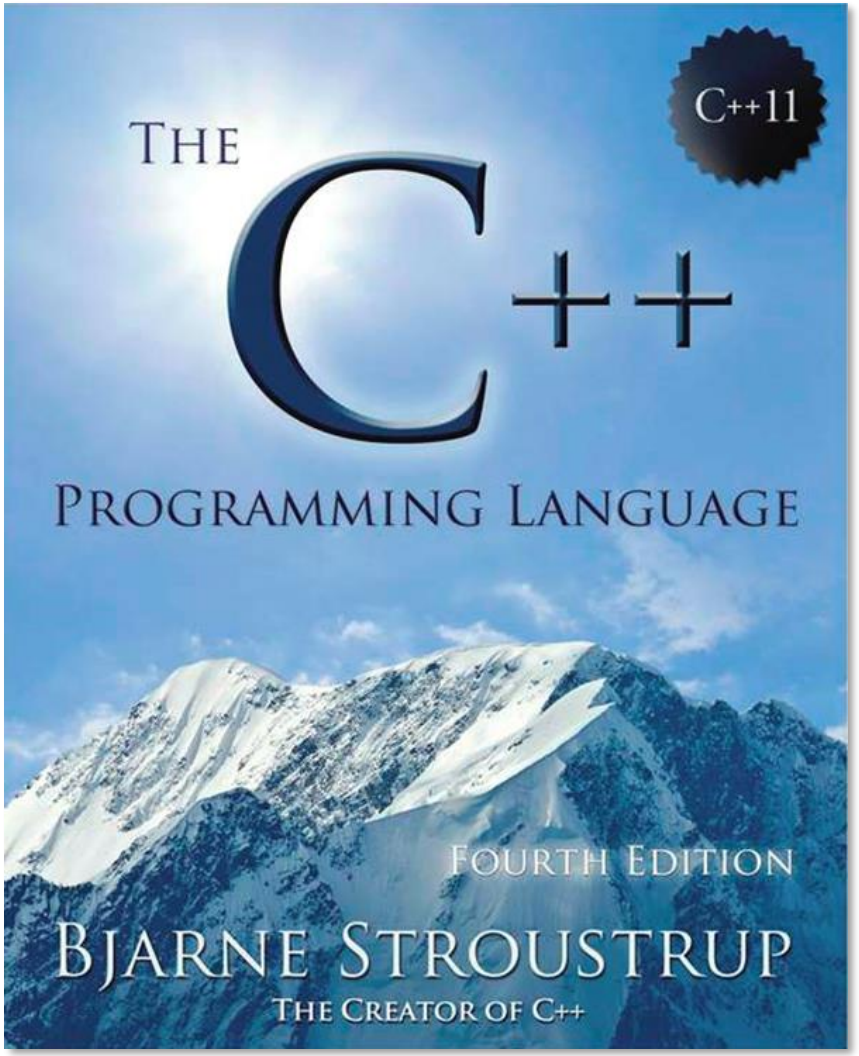
Professor of Computer Science

Institute for Software Integrated Systems

Vanderbilt University
Nashville, Tennessee, USA

Learning Objectives in this Part of the Lesson

- Recognize the key components of C++
- Know strategies for learning C++



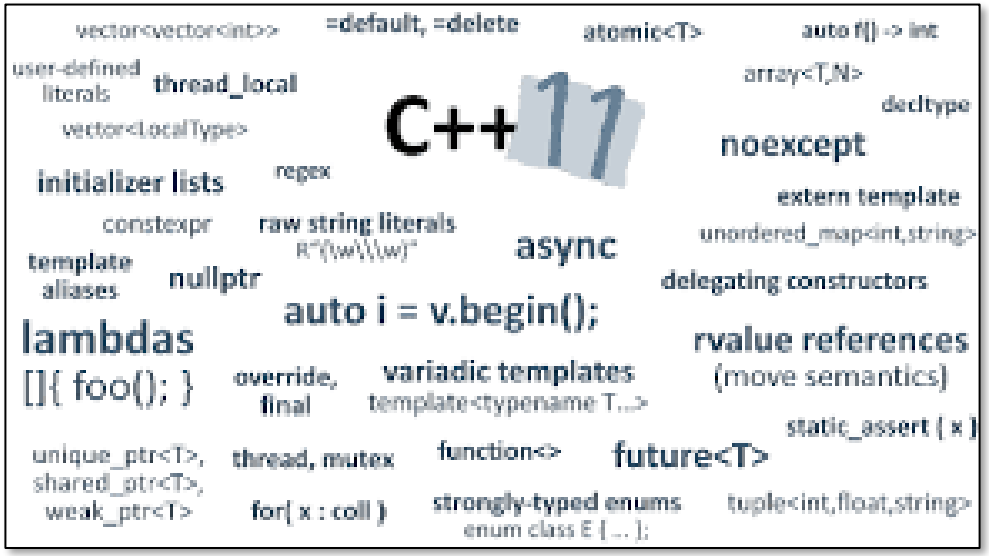
Strategies for Learning C++

Strategies for Learning C++

- C++ is a big language with a long history, so it's important to learn it smartly!

Strategies for Learning C++

- C++ is a big language with a long history, so it's important to learn it smartly!
  1. Focus on core concepts & design & programming techniques
     - Don't get lost in every new-fangled C++ language feature



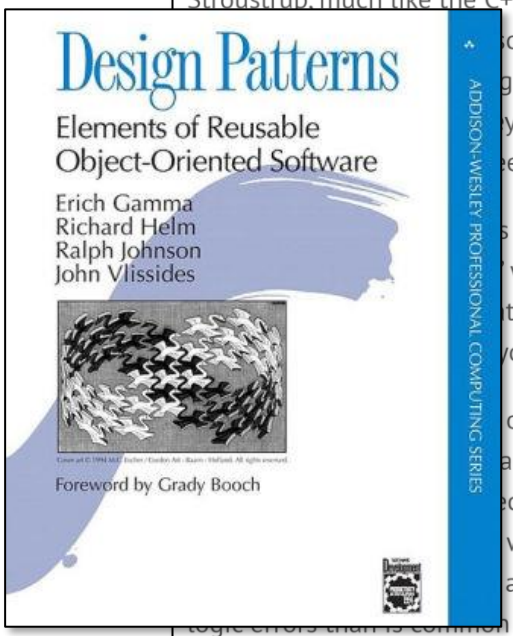
Many newer C++ features are intended for library developers, not app developers

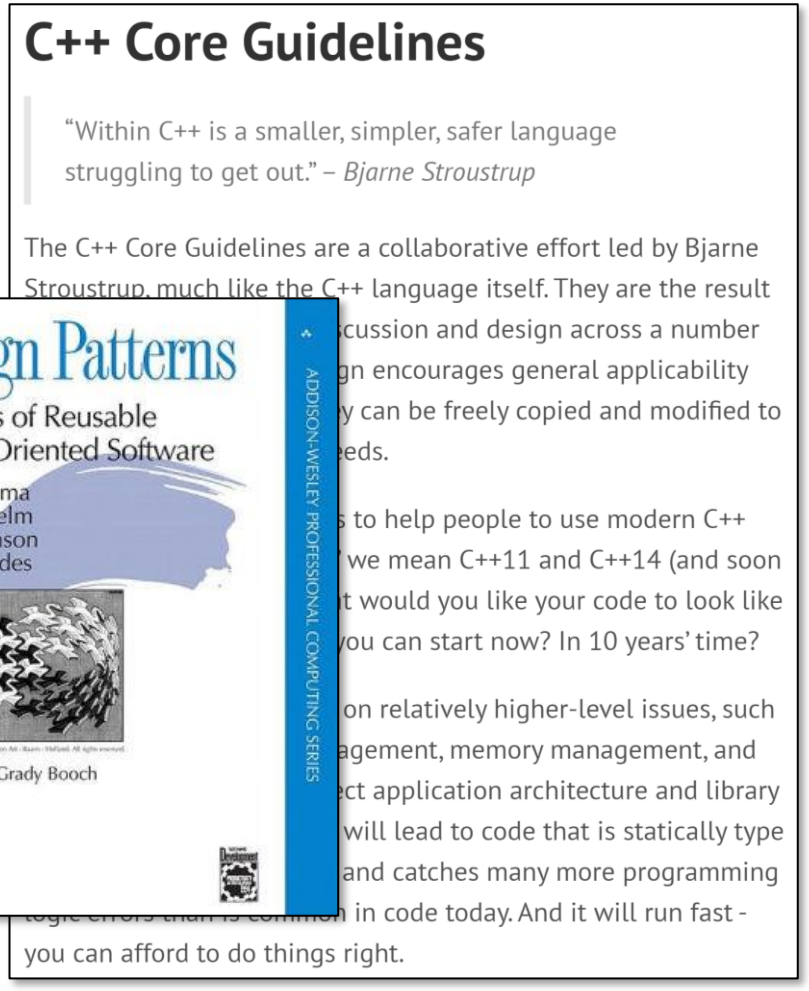
Strategies for Learning C++

- C++ is a big language with a long history, so it's important to learn it smartly!
  1. Focus on core concepts & design & programming techniques
  2. Learn C++ to become a better software developer
     - i.e., become more effective at developing & testing modular & robust programs

Strategies for Learning C++

- C++ is a big language with a long history, so it's important to learn it smartly!
  1. Focus on core concepts & design & programming techniques
  2. Learn C++ to become a better software developer
  3. Learn & apply software patterns & idioms
     - C++ supports many different programming styles

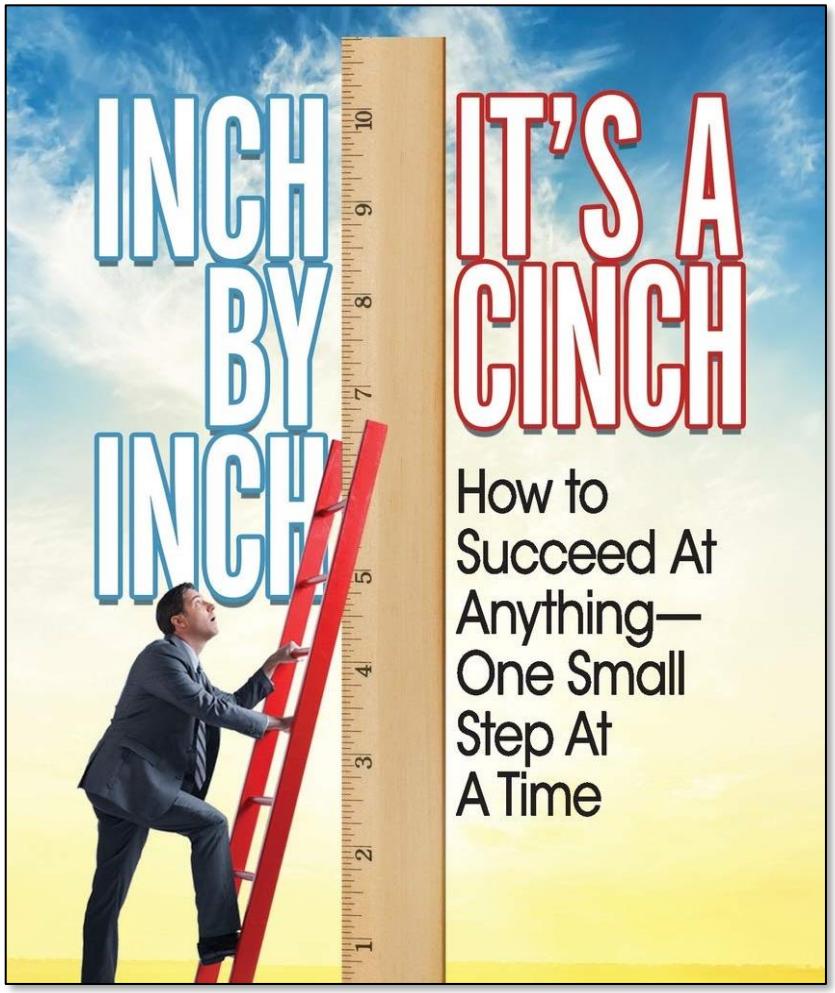




See en.wikipedia.org/wiki/Design_Patterns & isocpp.github.io/CppCoreGuidelines

Strategies for Learning C++

- C++ is a big language with a long history, so it's important to learn it smartly!
  1. Focus on core concepts & design & programming techniques
  2. Learn C++ to become a better software developer
  3. Learn & apply software patterns & idioms
  4. Learn C++ gradually
     - Don't have to know every detail of C++ to write good C++ programs



End of Overview of Strategies for Learning C++