

# CS100

# Introduction to Programming

Lecture 0

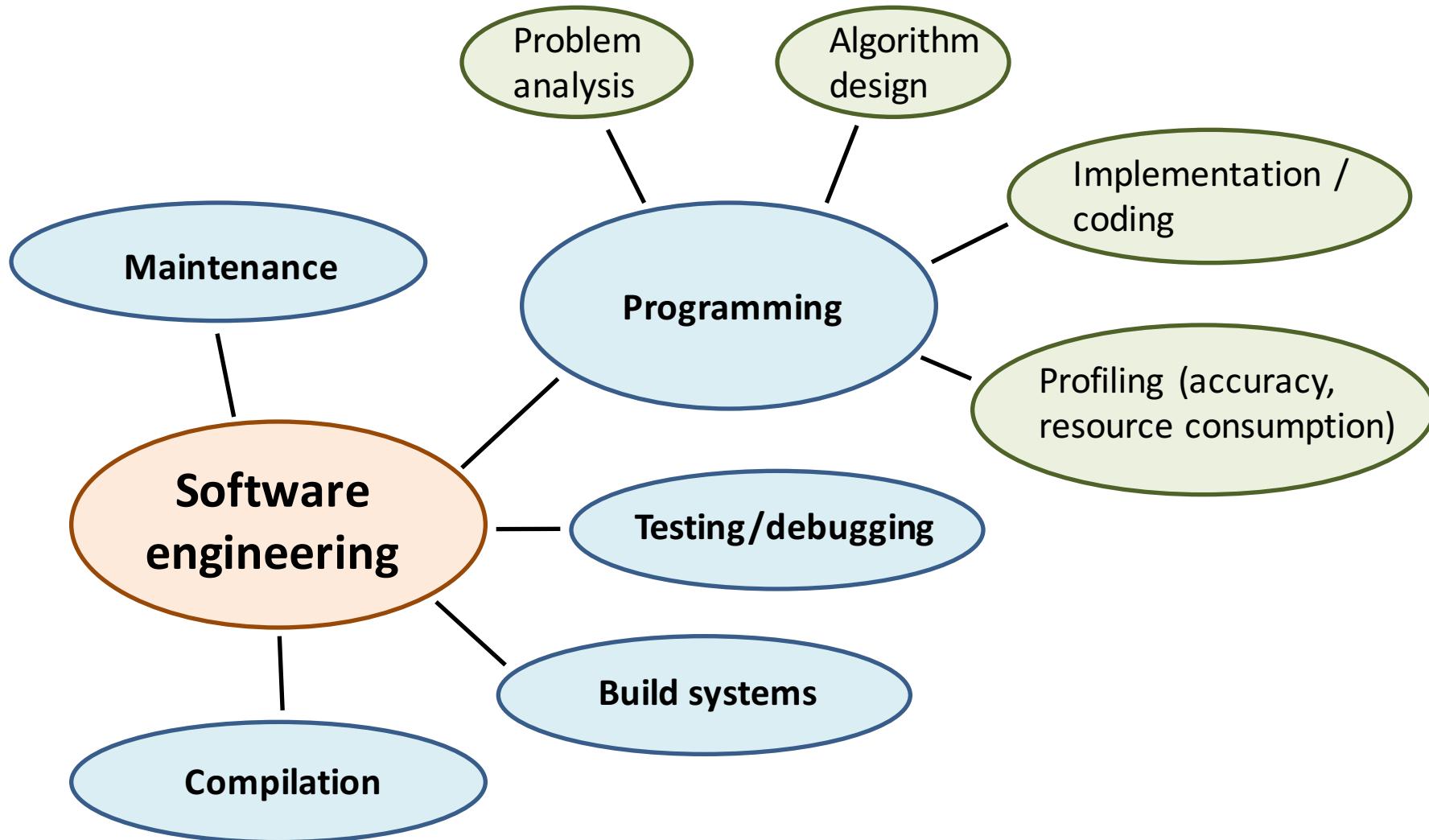
Course Introduction

# What is programming?

Computer **programming** is the process of writing instructions that get executed by computers. The instructions, also known as code, are written in a **programming language** which the computer can understand and use to perform a task or solve a problem.

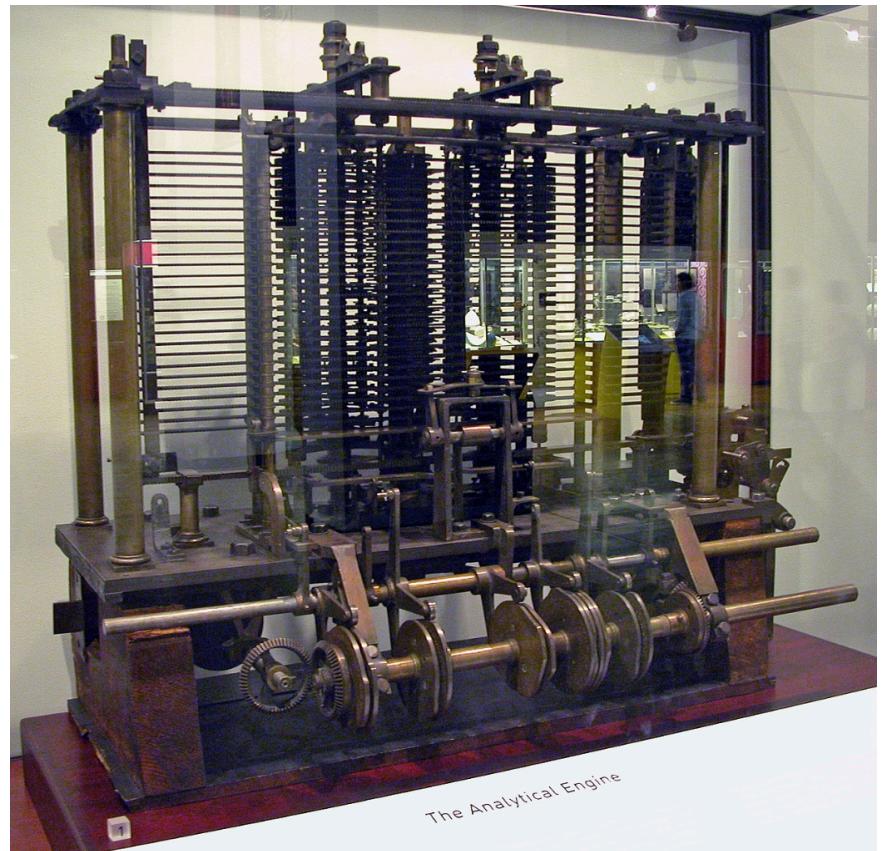
**Steve Jobs:** “*Everybody in this country **should learn** to program a computer, because it teaches you how to **think***”

# What is programming?



# A bit of history

- Analytical engine
  - Charles Babbage
  - Designed in 1837
  - Branching & Looping!
  - Turing-complete!



# A bit of history

- Analytical engine
  - Program is communicated via punched cards
  - Akin to modern day assembler language



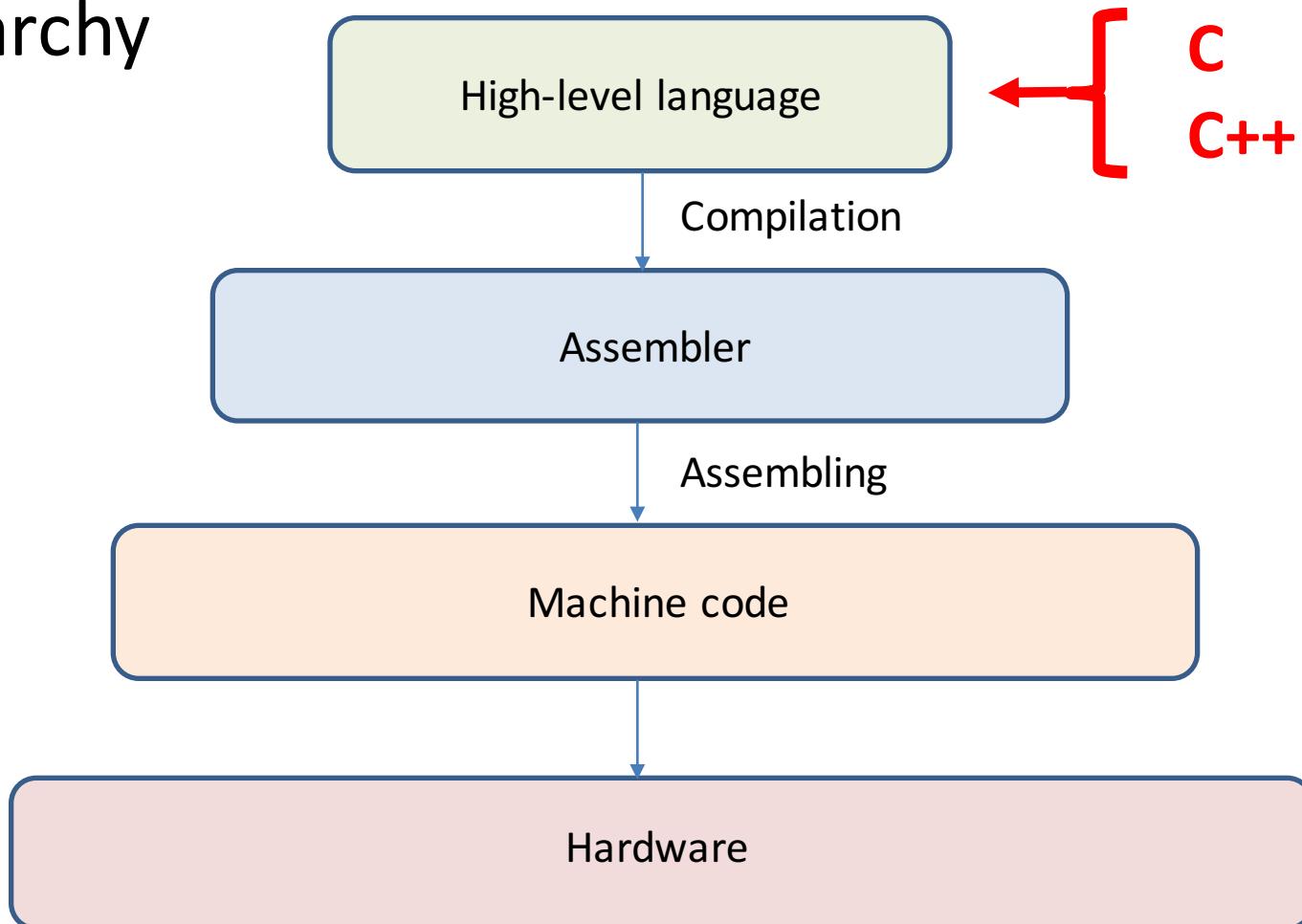
# Modern coding languages

- Syntactically rich
  - Better abstraction ability



# Modern coding languages

- Hierarchy



# Why Learning Programming in C?

- Why C?
  - Popular, early low-level language
    - Permits implementation of sequential programs
  - “Standard” syntax
- Why C++?
  - Popular higher-level language (Object-oriented)
  - Efficient programs

# Course Objectives

- Upon completion of the course, you should be able to:
  - Know how to analyze simple problems, and design programs
  - Know how to write them in C or C++
  - Know C/C++ syntax
  - Understand sequential vs. OO-programming
  - Get basics in functional programming
  - Have come in touch with multi-tasking and some useful C++ libraries

# Lecturers

**C**



**Xiaopei Liu**

Tel: (021) 20685382

Email:

[liuxp@shanghaitech.edu.cn](mailto:liuxp@shanghaitech.edu.cn)

**C++**



**Laurent Kneip**

Tel: (021)20685359

Email:

[lkneip@shanghaitech.edu.cn](mailto:lkneip@shanghaitech.edu.cn)

# Course Structure

- 15 weeks teaching (net)
  - 1 block C (5 weeks, 10 lectures)
  - 1 block C++ (9 weeks, 18 lectures)
  - 1 week midterm exam/reviews (approx.)
- 2 lectures / week
  - Wednesday 10:15-11:55
  - Friday 10:15-11:55
  - Room: Teaching Centre 201
- 1 weekly recitation
  - 16 groups of 17-18 students

# Course Page & Material

- Main course page:
  - <https://piazza.com/shanghaitech.edu.cn/fall2019/cs100/resources>
  - Will contain
    - Lecture slides (including these slides)
    - Recitation material
    - Annoucements
    - Homeworks
    - Q&A forum
  - **Please enrol by yourself!**

# Tentative curriculum

- Lecture 1: Intro/C program structure
- Lecture 2: Memory, simple I/O and control flow
- Lecture 3: Functions and pointers
- Lecture 4: Arrays
- Lecture 5: Character strings
- Lecture 6: Structures
- Lecture 7: Recursion
- Lecture 8: Summary of C and advanced C
- Lecture 9: Classes and inheritance
- Lecture 10: Polymorphism

# Tentative curriculum

- Lecture 11: Object-oriented programming with C++
- Lecture 12: Introduction to C++ operators and STL
- Lecture 13: STL templates
- Lecture 14: STL data structures
- Lecture 15: Review towards mid-term exam
- Lecture 16: **Midterm exam**
- Lecture 17: Memory management in C++, shared pointers
- Lecture 18: Structuring your code (part 1)
- Lecture 19: Structuring your code (part 2)
- Lecture 20: Examples

# Tentative curriculum

- Lecture 21: Code development: Coding standards
- Lecture 22: Code development: CMake
- Lecture 23: Code development: Large projects
- Lecture 24: Code development: Profiling/Debugging
- Lecture 25: C++11: Concurrency
- Lecture 26: C++11: Rvalue references
- Lecture 27: C++11: Functional programming
- Lecture 28: New features of C++
- Lecture 29: Design patterns
- Lecture 30: Review

# Recitations

- 13 Units
- 2 Hr/week
- Individual groups of ~17 students, each having a fixed TA throughout the semester
- Recitations are held on a weekly basis, and there will be 16 different groups @ different times
- Enrolment information will be posted on Piazza  
<https://piazza.com/shanghaitech.edu.cn/fall2019/cs100/home>
- **Attendance mandatory!**

# Recitations

- A mix of:
  - Revision of the course material
  - Provision of extra examples
  - Introduction of some new material & examples
  - Interactive examples:
    - Students are asked to solve some problems in class
    - Answers provided during recitation
  - Clarification of general questions related to homeworks
  - Introduction of homework solutions
- Additional material (i.e. slides) will be posted on Piazza
- Recitations will be starting from 3<sup>rd</sup> week (23<sup>rd</sup> of September)

# Homework assignments

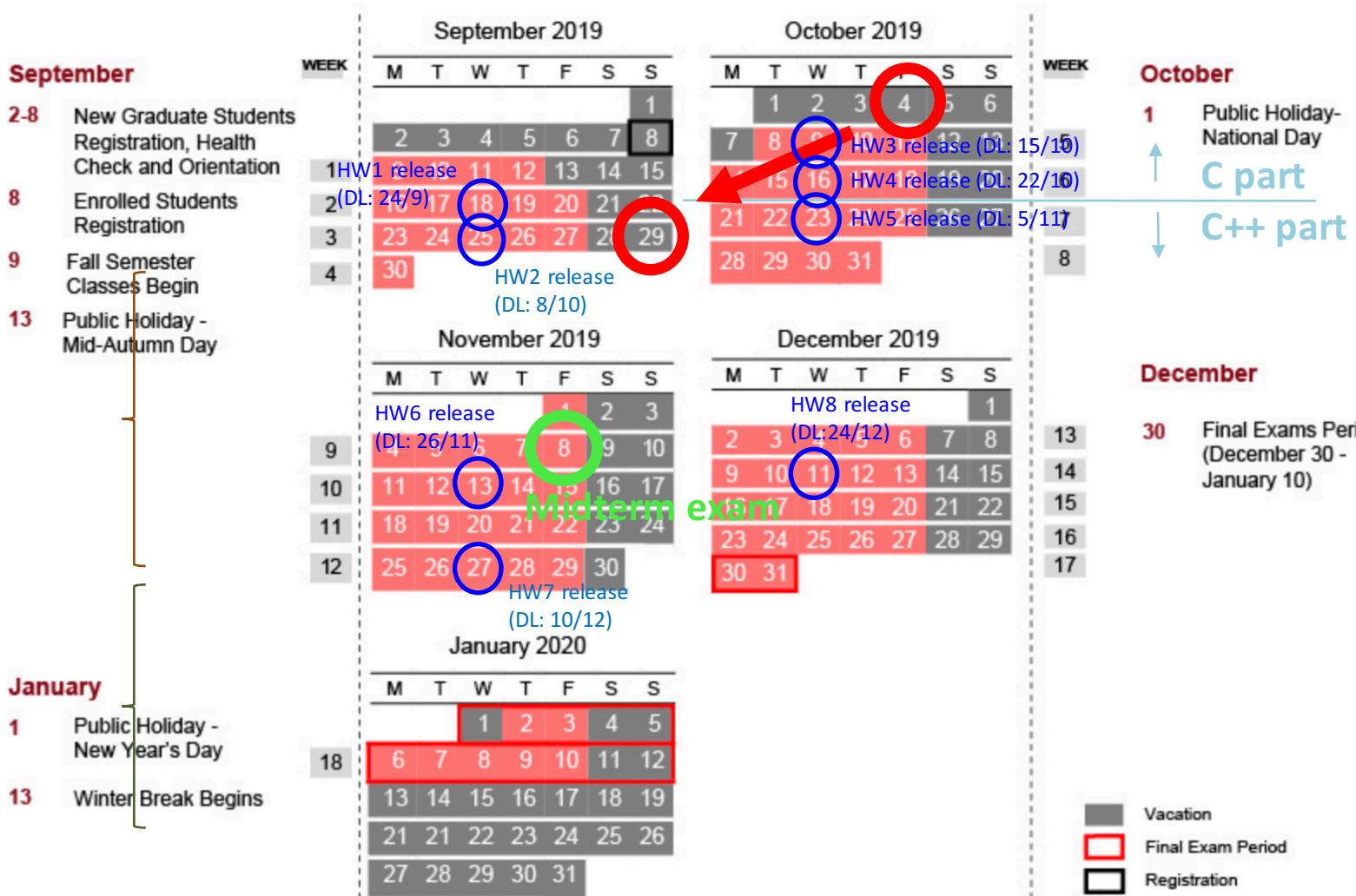
- 8 homeworks:
  - 4 shorter homeworks primarily on C at the beginning
    - weekly
  - 4 longer homeworks afterwards after mid-term exam
    - Bi-weekly
- Makes up for large part of your grade!
- Will be released during Wednesday lectures
- Solution will be reviewed during recitations
- Strict deadline ~7/14 days later, 23:59 on the day!
- Late submission policy:
  - 50% penalty if submitted before 23:59 on the day following the deadline
  - 100% penalty if no submission by 23:59 on the day following the deadline

# Homework submissions

- Through Online Judge
  - Simple OJ used during beginning
    - Same OJ as used during summer homeworks
  - A github based OJ may be used later if required
    - After introduction to gitlab in class
- **We will check the code manually as well!**

# Timeline

- 16 weeks! (but free on 13<sup>th</sup> of Sep. & 2<sup>nd</sup> of Oct.)



# Summer homeworks

- Deadline
  - 20<sup>th</sup> of September 23:59PM!
  - No late submission policy!

# Quizzes

- 2 Quizzes
- During regular class, probing class attendance
- Time will not be announced in advance!
- Sickness policy:
  - Requires medical certificate
  - Requires attendance to make-up session  
(last chance!)

# Attendance check of recitations

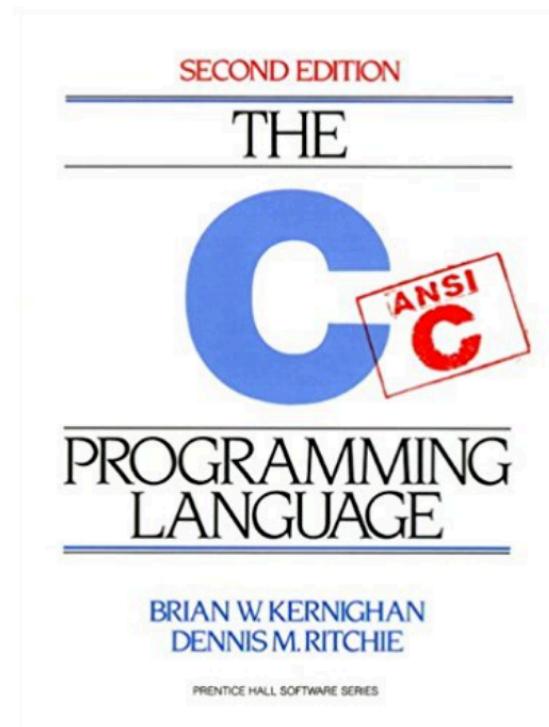
- We will also probe the attendance of the recitations at two random points during the semester. It would be done simply by collecting signatures.

# Assessment

- A mix of
  - Summer homeworks: 4%
  - Regular homeworks:
$$\begin{aligned} & 4 \times 7\% + 4 \times 11\% \\ & = 28\% + 44\% \\ & = 72\% \end{aligned}$$
  - Mid-term exam: 18%
  - In-class quizzes: 4%
  - Recitation attendance checks: 2%
- **No final Exam!**

# Recommended Literature

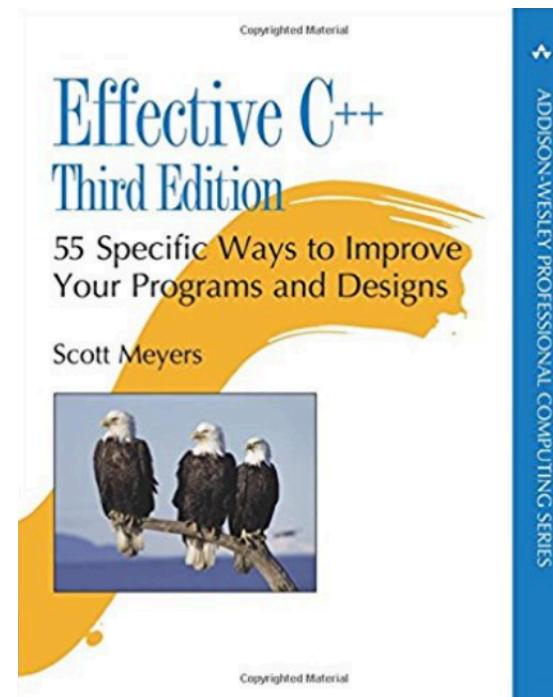
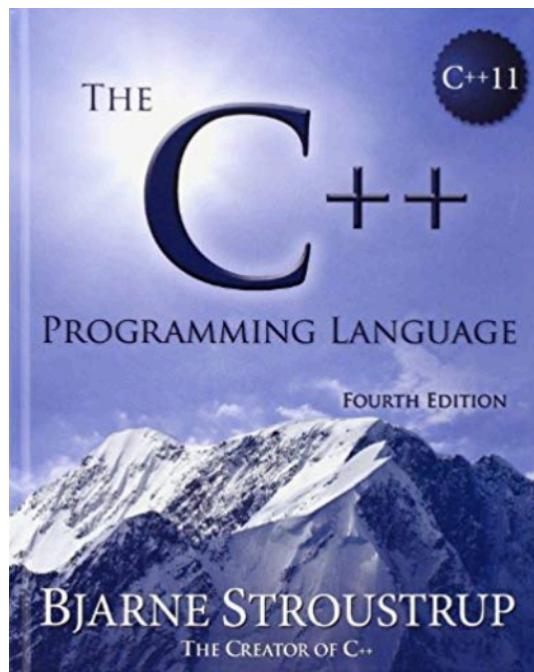
- C
  - *The C Programming Language* (2nd Edition), by Brian W. Kernighan and Dennis M. Ritchie, Prentice Hall, 2014 (available in the library)



# Recommended Literature

- C++

- *The C++ Programming Language*, 4th Edition, Bjarne Stroustrup, Pearson, 2013
- *Effective C++*, 3rd Edition, Scott Meyers, Addison-Wesley, 2005



# Recommended Literature

- General note:
  - You are not required to buy those books
  - These are only suggestions in case you feel like buying a book on this
  - Our material is independent and will contain all the material

# Communication

- Your primary point of contact
  - Your TA!
  - For all course material related matters
- Use the Piazza forum!
  - However, no code sharing of homeworks!
  - You may post general questions about language or course organization
  - We encourage you to first Google yourself and talk to TAs

# Communication

- Teacher's contact
  - Prof Xiaopei Liu:
    - Email: [liuxp@shanghaitech.edu.cn](mailto:liuxp@shanghaitech.edu.cn)
    - Office hours: TBD
  - Prof Laurent Kneip:
    - Email: [lkneip@shanghaitech.edu.cn](mailto:lkneip@shanghaitech.edu.cn)
    - Office hours: Wednesdays, 2pm-3pm, SIST 1C-303E

# Academic Integrity

- Unless explicitly noted, work turned in should reflect your own/independent capabilities
- No cheating
  - Do not share your homework/gitlab PW under any circumstances! Always protect your homework!
  - No “fake solutions” (**we will check!**)
  - **No plagiarism** (copying of part/complete solution from somewhere else) (**we will check!**)
  - **Serious consequences, including the possibility of being expelled!**