



COMP2012H Honors Object-Oriented Programming and Data Structures

Topic 0: Course Logistics

Dr. Desmond Tsoi

Department of Computer Science & Engineering
The Hong Kong University of Science and Technology
Hong Kong SAR, China



Rm 3553, desmond@cse.ust.hk

COMP2012H (Fall 2018)

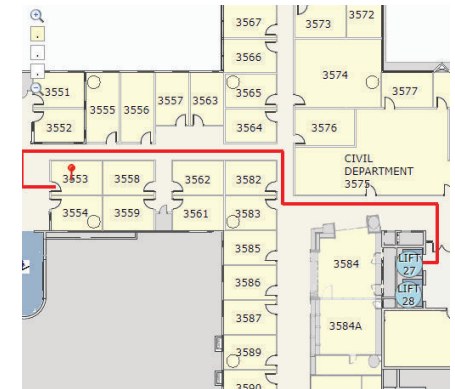
1 / 29

Instructor

Dr. Desmond Yau-chat TSOI (Simply call me "Desmond" ;))



- Personal website:
<http://www.cse.ust.hk/~desmond>
- E-mail: desmond@cse.ust.hk
- Office: Rm 3553 (Lift 27)
- Work phone: 2358-6984
- Office hours: To be confirmed



Rm 3553, desmond@cse.ust.hk

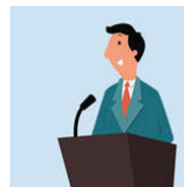
COMP2012H (Fall 2018)

2 / 29

More about Me

List of institutes that I was teaching at: (in chronological order)

- Hong Kong University of Science and Technology (HKUST)
 - Department of Computer Science and Engineering
 - Department of Accounting
- Nanyang Technological University, Singapore (NTU)
 - Department of Computer Science, School of Computer Engineering
- Hong Kong College of Technology
 - Department of Computer and Information Technology (HKCT)
- Community College of City University (CCCU)
 - Division of Applied Science and Technology
- Hong Kong Polytechnic University
 - School of Professional Education and Executive Development (SPEED)



Now, I am once again serving CSE, HKUST

Note

You are welcome to talk to me if you have any questions about further study and / or career development!

Rm 3553, desmond@cse.ust.hk

COMP2012H (Fall 2018)

3 / 29

Teaching Assistants

Postgraduate TAs

- CHANG, Bing Yen
 - E-mail: bychang@connect.ust.hk
 - Office hours: To be arranged



Rm 3553, desmond@cse.ust.hk

COMP2012H (Fall 2018)

4 / 29

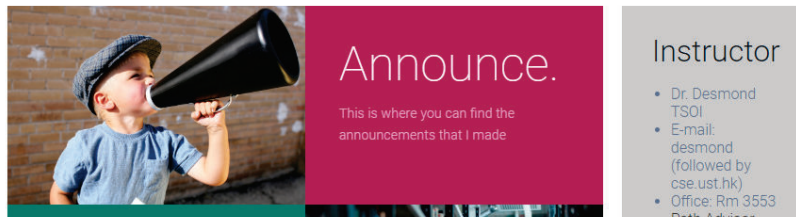
Course Website and Supplementary Website

- Course website:
<https://course.cse.ust.hk/comp2012h>
- Supplementary course website:
<http://desmond.people.ust.hk/comp2012h>
 - ▶ Login: comp2012h
 - ▶ Password: < please mark it down :) >

COMP 2012H Honors Object-Oriented Programming and Data Structures (Fall 2018)

Supplementary Site

Official Site: <https://course.cse.ust.hk/comp2012h/>



Rm 3553, desmond@cse.ust.hk

COMP2012H (Fall 2018)

5 / 29

Lectures

- Section L1 (Dr. TSOI, Desmond)
Tuesday and Thursday,
09:00am - 10:50am,
Rm 2406 (Lift 17-18)



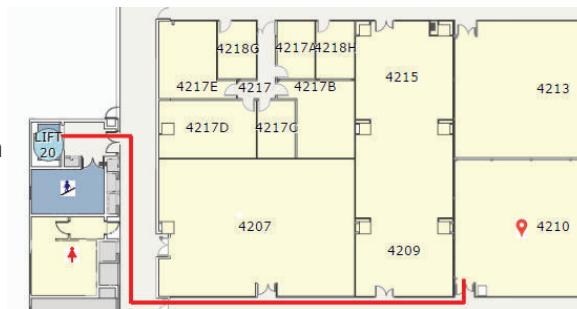
Rm 3553, desmond@cse.ust.hk

COMP2012H (Fall 2018)

6 / 29

Labs (Rm 4210, Lift 20)

- Section LA1
Wed, 09:00am - 10:50am
- Section LA2
Thu, 07:00pm - 08:50pm



Note

1. No swapping of labs without the approval from the instructor
2. There will be no lab on the first two weeks
 - ▶ The first lab will be conducted in week 3 (September 17-21)

Rm 3553, desmond@cse.ust.hk

COMP2012H (Fall 2018)

7 / 29

Course Description

- The course consists of, per week
 - ▶ 4 hours of lectures
 - ▶ 2 hours of lab exercisesand it gives 5 credits for successful completion of the course.
- Prerequisites
Grade A or above in
 - ▶ COMP 1002 Computer and Programming Fundamentals I (prior to 2013-14) OR
 - ▶ COMP 1021 Introduction to Computer Science OR
 - ▶ COMP 1022P Introduction to Computing with Java OR
 - ▶ COMP 1022Q Introduction to Computing with Excel VBA OR
 - ▶ ISOM 3230 Business Applications Programming
- Exclusion
 - ▶ COMP 1004 Programming Fundamentals & Methodology (prior to 2013-14)
 - ▶ COMP 2011 Introduction to Object-Oriented Programming
 - ▶ COMP 2012 Object-Oriented Programming and Data Structures

Rm 3553, desmond@cse.ust.hk

COMP2012H (Fall 2018)

8 / 29

Course Objectives / Aims

- This is an accelerated and intensive course on concepts and techniques behind object-oriented programming (OOP) and data structures using an OOP language.
- It covers the major materials of COMP2011 and COMP2012, and its curriculum is designed for students with excellent programming background or substantial programming experience.



In short

COMP2012H is a highly accelerated course, “not depth”.

Course Objectives / Aims

- Topics include:
 - introduction to computer programming;
 - fundamentals of C++:
 - data types, variables, operators
 - flow controls;
 - arrays;
 - functions, scope, and recursion;
 - file I/O;
 - pointers;
 - structures and linked lists;
 - class, object construction, initialization, and destruction;
 - inheritance and polymorphism;
 - generic programming;
 - rvalue reference and move semantics;
 - Standard Template Library;
 - static data members and member functions;
 - stack and queue;
 - binary search trees and AVL trees;
 - hashing.



Intended Learning Outcomes

On successful completion of this course, you are expected to be able to:

1. Use common software tools to develop and debug a program written in an OOP language.
2. Demonstrate that recursive and non-recursive functions are abstractions of sub-problems in a task.
3. Describe the concept and the use of pointers in indirect addressing and dynamic memory allocation.
4. Write object-oriented programs in C++ with object creation, destruction, member variables and functions, inheritance, polymorphisms, and templates



Intended Learning Outcomes

On successful completion of this course, you are expected to be able to:

5. Analyze a program and provide simple solutions with OOP
6. Write basic algorithms associated with data structures such as stacks, queues, lists, trees, and hashes
7. Define binary tree and search tree and describe how they are used to solve problems
8. Develop a large program using separate compilation, good OOP design, and code reuse.



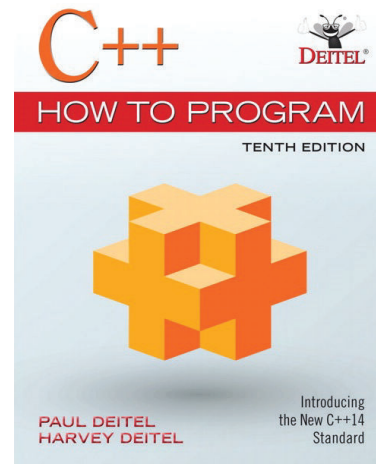
Recommended Textbooks

- C++ How to Program, Paul J. Deitel and Harvey M. Deitel & Associates, Pearson, c2017, Tenth Edition.

Available at university bookstore

- ▶ UST library for 2014 version
 - ★ QA76.73.C153 D45 2014
 - ★ QA76.73.C153 D45 2014 c.2

ISBN: 0133378713 (1028 pages)



Note

- ★ This textbook uses C++11 codes
- ★ Discusses about new features in C++14

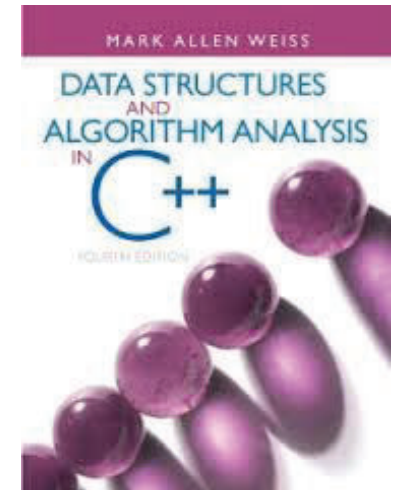
Recommended Textbooks

- Data Structures and Algorithm Analysis in C++, Mark Allen Weiss, Pearson, c2014, Fourth Edition.

- ▶ QA76.73.C153 W46 2014
- ▶ QA76.73.C153 W46 2014 c.2

ISBN: 013284737X (635 pages)

Available at HKUST Library and university bookstore



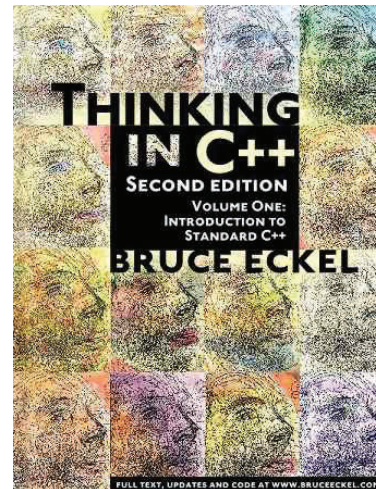
Recommended Textbooks

- Thinking in C++, Eckel Bruce, Prentice Hall, c2000 - c2004, Second Edition.

- ▶ QA76.73.C153 E247 2000 v.2
- ▶ QA76.73.C153 E247 2000 v.2 c.2

ISBN: 0139798099

Available at HKUST Library.



A free copy from the publisher is available online at:
<http://mindview.net/Books/TICPP/ThinkingInCPP2e.html>

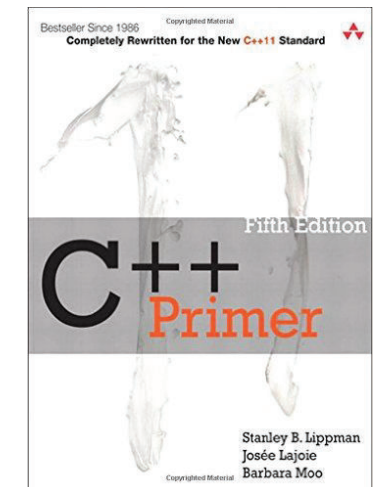
Recommended Textbooks

- C++ Primer, Stanley B. Lippman, Josee Lajoie, Barbara E. Moo, Addison-Wesley, c2013, Fifth Edition.

- ▶ QA76.73.C153 L57697 2013

ISBN: 0321714113 (938 pages)

Available at HKUST Library.



Other Materials

- Self-tests
- Past COMP 2011 & COMP 2012 exam papers
- Reserved books in HKUST library
 - ▶ The C++ Programming Language
 - ★ Author: Bjarne Stroustrup
 - ★ Call #: QA76.73.C153 S77 2013
 - ▶ Problem Solving with C++
 - ★ Author: Walter Savitch
 - ★ Call #: QA76.73.C153 S273 2012
 - ▶ Programming in C++: Lessons and Applications
 - ★ Author: Timothy B. D' Orazio
 - ★ Call #: QA76.73.C153 D66 2009
 - ▶ C++ for Everyone
 - ★ Author: Cay S. Horstmann
 - ★ Call #: QA76.73.C153 H6685 2012



Bjarne Stroustrup
[Pronunciation]
Designer and original
implementor of C++



Alan Kay
Pioneer of OOP

Tentative Teaching Schedule

Topic	#Lectures	Cumulative #Lectures
Introduction	0.5	0.5
Fundamentals of C++	0.5	1
Flow Controls	1	2
Functions, Scope, and Recursion	1	3
Arrays	1	4
Pointers	2	6
Structures and Linked List	2	8
Class, Object Constr., Init., & Destruction	2	10
Inheritance and Polymorphism	3	13
Generic Programming	3	16
rValue Reference and Move Semantics	Additional Class	16
Standard Template Library	2.5	18.5
Static Data Members and Member Functions	1	19.5
Stack and Queue	1	20.5
Binary Search Tree (BST)	2	22.5
AVL Tree	2	24.5
Hashing	1.5	26

Note

- The schedule is subject to change according to the teaching and learning progress!
- Makeup / Additional classes will be offered
 - ▶ September 25 (Tuesday): The day following the Chinese Mid-Autumn Festival

C++ Version and Software

- We use C++ version 2003 & 2011 as the programming standard for this course
- Integrated Development Environment (IDE)
 - ▶ Eclipse (version 4.6.2) Neon.2 with MinGW/GCC version 6.3
- C++ Compiler
 - ▶ g++



Note

- Starting from 2011, C++ rolls out a new standard every 3 years!
- Compilers need time to catch up
- You are suggested to check your compilers if they support the new codes

Grading Scheme

Assessment breakdown:

- Coursework (50%)
 - ▶ 9 Lab Exercises (10%)
 - ▶ 4 Individual Programming Assignments (28%)
 - ▶ 1 Group Programming Project (12%)
- Examination (50%)
 - ▶ Mid-term Exam (20%)
 - ▶ Final Exam (30%)



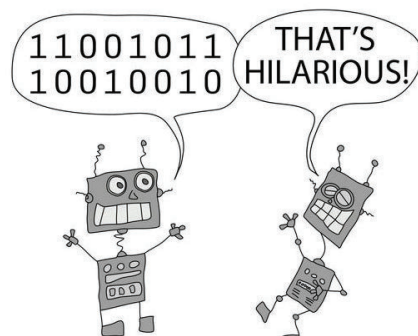
Coursework (50%) + Examination (50%) = Total (100%)

Note

No make-up exams will be given unless under very unusual circumstances, e.g., sickness, with letters of proof

Lab Exercises

- Programming exercises will be given to consolidate your understanding of course materials.
- 3 points for each lab
 - ▶ 1 point for **attendance**
 - ▶ 1 point for **completing your work and demonstrating it** to a TA / student helper during lab time
 - ▶ Answer a **question** about the lab exercise if you are selected. If you are not selected, you have it on completion of the above two
- Materials will be released a week before the lab.



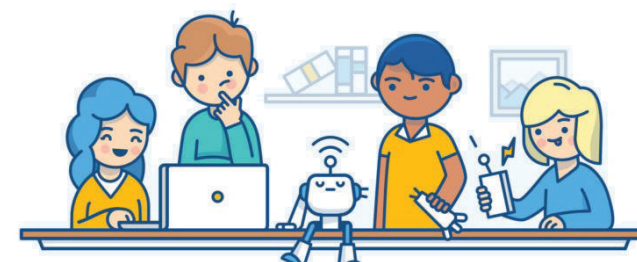
Individual Programming Assignments

- Tentative schedule

	Release Date	Due Date
PA1	September 17 (Monday)	October 1 (Monday)
PA2	October 1 (Monday)	October 15 (Monday)
PA3	October 15 (Monday)	October 29 (Monday)
PA4	October 29 (Monday)	November 12 (Monday)

Code Submission

Submission should be done on [Canvas](#)



Group Programming Project

- Form a group of 2 to 3 people to work on a self-proposed project.
- Project deliverables
 - ▶ Proposal report (one A4 page) describing what problem(s) you try to address and state why you find the problem(s) important or interesting
 - ▶ Code with documentation
 - ▶ Project Demonstration
- Tentative schedule

Group Forming Deadline	Proposal Submission Deadline	Project Submission Deadline	Project Demonstration
October 29	November 12	November 26	November 28 or 29 (Depending on your lab section)

Proposal and Code Submission

Submission should be done on [Canvas](#)

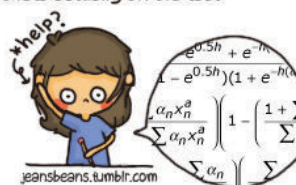
Midterm Exam

- The midterm examination is scheduled on
October 20, 2018 (Saturday), 10am - 12nn
- Venue: Lecture Theater F

what we learn in class
and what i study



whats actually on the test



jeansbeans.tumblr.com

Academic Dishonesty

- Anyone “both the copier and the copier” caught cheating on their assignments will get zero marks for the assignment.
- In addition, they BOTH will **get an additional 10%** deduction from the final grade for the first time of cheating. Anyone get caught for cheating in an assignment for the **2nd time** will get an immediate **FAIL** grade.
- On the other hand, anyone caught cheating in the midterm or final exam will get a FAIL grade **immediately**.
- There can be additional disciplinary actions as well from the department and university.
- Links:
 - ▶ University's Honor code:
<http://ugadmin.ust.hk/integrity/student-1.html>
 - ▶ University's Penalties for Cheating:
<http://ugadmin.ust.hk/integrity/student-5.html>

Academic Dishonesty (Cont'd)

- We will use a software to check your codes with others' program, and even with previous assignments. The tool is hard to beat. The suspected cases will be further studied by the instructor and the TAs.



If you are not sure what is considered plagiarism

- **DO NOT** copy program codes from another student/person.
- **DO NOT** look at the actual program codes of another student.
- **DO NOT** share actual program codes with other students/people (by paper, emails, blogs, FB, Google Doc, etc.).
- **DO NOT** give your program codes to other students who ask for it, and do not ask for a copy of their code either.
- **DO NOT** post your program codes anywhere online.
- **DO NOT** leave your finished/unfinished program codes unattended.
- While we encourage discussion among students, you have to write codes on your own.
- During discussion, you **SHOULD NOT** go to the details such that everyone will end up in the same code.

How Hard Should I Work?

- Some people say that a 3-unit course takes 8 hours/week.
- For COMP 2012H, it is a **5-unit course**, so you are expected to spend more time, e.g., **13-14 hours/week**.
- Guideline:
 - ▶ Pre-study (1 hour): what topic/materials will the coming lecture be covering?
 - ▶ Attend class (4 hours): The A+ students tell you that they pay FULL attention in class and try to understand everything in the class so that it is easy to review the class materials.
 - ▶ Attend labs (2 hours)
 - ▶ Post-study (8-9 hours): Re-reading the notes, book reading.



That's all!
Any question?



**Welcome
Back!**