



COMP2511

Object-Oriented Design and Programming

Introduction

Wayne Wobcke

w.wobcke@unsw.edu.au



Second Year is not First Year

- What did you learn in first year?
 - ◆ And Java is not C
 - ◆ And this is not a Java course
 - ... but you have to write good Java!



Today's Lecture

- Second Year is not First Year
- Design
- Software Engineering
- Course Outline
- Java and Eclipse



Design

- What is design?



Software “Engineering”?

■ What is good engineering design?

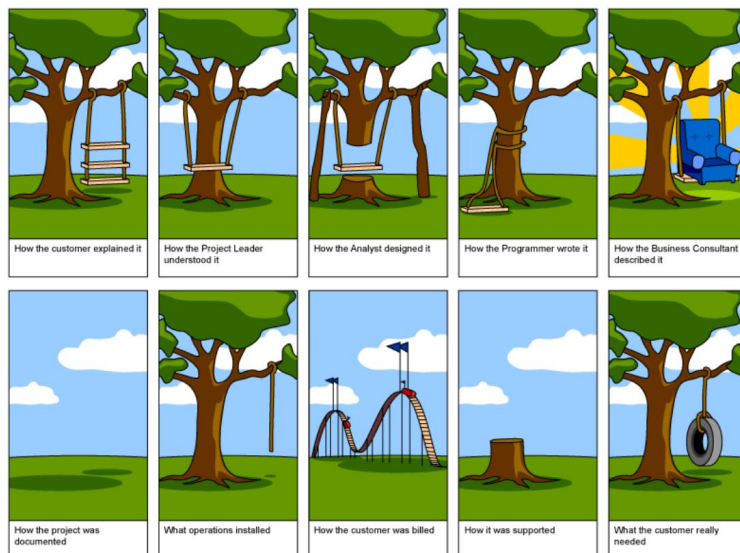
- ◆ Purpose?
- ◆ Reliability?
- ◆ Quality?
- ◆ Aesthetics?
- ◆ Materials?
- ◆ System structure?



Coding

■ ... **not** the most important aspect of producing high quality software

- ◆ Engineering/Design
- ◆ Doing it properly – Best practice(s)
- ◆ Correctness not enough
- ◆ Design **before** coding
- ◆ Teamwork and communication
- ◆ Understanding the users



Toyota Production System

■ Today produce software like cars

- ◆ Focus on end users, business value
- ◆ Elimination of waste (over-engineering)
- ◆ People over process
 - But don't ignore process!
- ◆ Kaizen (continuous improvement)
- ◆ Kanban (pull/signal)
- ◆ **Not just a collection of tools/processes**



Course Outline

■ Key points


- ◆ Two cohorts, half done COMP1531
- ◆ Course runs in both semesters
- ◆ Knowledge/skills incremental and build on **both** lectures and labs (agile)
- ◆ Blended learning – Read beforehand!
- ◆ Tut-labs focus on best practice(s)
- ◆ Use supplied references (not Internet)
- ◆ Plagiarism (keep github repos private)



Java algorithm

Status: **Open** | Posted on: 13/05/2013
 Project budget: **\$100.00**
 Project ID: 43572


[PLACE BID](#) [VIEW MESSAGE BOARD \(13\)](#) [WATCH](#)

 **Buyer cannot cover the payment on-site.**
 Posted by: [mich](#)
 Time remaining: 3 day(s)
 Tags: [Java](#)

Project description:

Please read the description here. easy money!
<http://www.cse.unsw.edu.au/~cs2911/13s1/assignments/ass02.html>

[PLACE BID](#) [VIEW MESSAGE BOARD \(13\)](#)

Freelancer	Total	Deliver in	Date of bid	Rating
 htt210	\$50.00	2 days	15/05/2013	Not rated

Dear customer, this assignment is very easy and can be done very quickly. I'm very good at Data Structure and Algorithm and Object Oriented Programming so your assignment will be perfect.



How almost every Stack Overflow question looks to me [I agree](#)

How do I do this thing?

43 [coding](#) [question](#)
 share | improve this question
 edited Apr 2 '12 at 8:13 [Grammar Nazi](#) 2.5M • 7 • 35 • 70
 asked Feb 1 '10 at 16:27 [1337z0r](#) 1337z0r • 2 • 1 • 3 • 6

5 Answers [active](#) [oldest](#) [votes](#)

12 [share](#) | improve this answer
 That's perfect! I'm never checking back here again! – [1337z0r](#) Feb 1 '10 at 16:42

248 [share](#) | improve this answer
 A lot of people think you should do it in a lazy way, however in the long run it will help you if you read this well eloquent wall of text that acutely describes problems you will inevitably face but not take the time to read about here, enjoy these code samples and illustrations I pulled from thin air anyway!
 answered Feb 1 '10 at 16:30 [1337z0r](#) 1,230 • 3 • 14 • 25

18 [share](#) | improve this answer
 The official way to do this is link.
 answered Feb 1 '10 at 16:29 [Professional Coder](#) 5,241 • 2 • 24 • 63

Uh. No thanks, this is too hard. Can you give me an example of how my code should look when complete? – [1337z0r](#) Feb 1 '10 at 16:41

Community Bulletin
 event Microsoft can help you port your app to Windows 8 + win prizes with AppInventor! – now through June 7

Work. From Home.
 If you are a dog...

CAREERS 2.0
 There really are jobs for coders available! Not for you though. Kersted Systems Houston, TX / relocation
 Highly paid, competitive benefits; that one language you didn't learn.



Course Outline

■ Improvements for 2018

- ◆ Course forum
 - General/assignment questions
 - Be wary of Facebook group!
- ◆ Resources guides for each week
 - What to focus on before and after
- ◆ Team project
 - More rigorous agile methods
 - Moodle Team Evaluation tool



Java and Eclipse

■ OOP & Java basics for Tut-lab 2

- ◆ JDK, JRE, JVM, API, IDE
- ◆ Classes and objects (one class per file)
- ◆ Inheritance (class extends another)
- ◆ Encapsulation (public and private)
- ◆ Java SE 8 API (e.g. Calendar)
- ◆ Eclipse IDE
- ◆ Javadoc



Next Week

■ Readings mix foundations & practice

- ◆ Object-oriented programming
- ◆ OOP vs ADTs
- ◆ Liskov Substitution Principle
- ◆ Java technicalities: equality, exceptions