

Software Construction

Brian Crites

Welcome to Upper Division

Purpose of this Course

CS 100
+
CS 180

A **software architect** is a software expert who makes high-level design choices and dictates technical standards, including software coding standards, tools, and platforms.

+

All Other
Classes

A **full stack developer** is an engineer who can handle all the work of databases, servers, systems engineering, and clients.

Goals

- Development Methodologies
- Testing Strategies/Methods
- System Design
- Common Tools
- Common Terminology
- “Large” Scale Development Practice
- Working with a Group
- Ethics

After this course you should be ready for an internship
and you **should apply** for one now!

UC Riverside

Rank #85 in National Universities

Rank #87 in Best Undergraduate Engineering
Programs

Rank #43 in Best Undergraduate Teaching

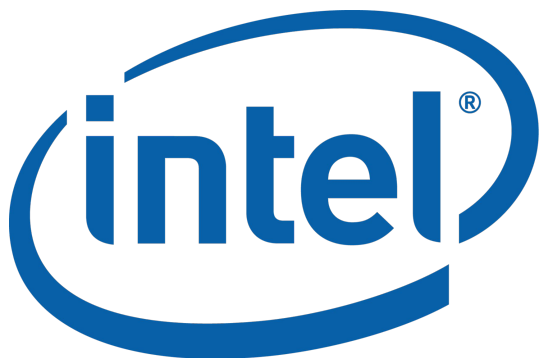
Rank #35 in Top Public Schools





Raytheon

NORTHROP GRUMMAN



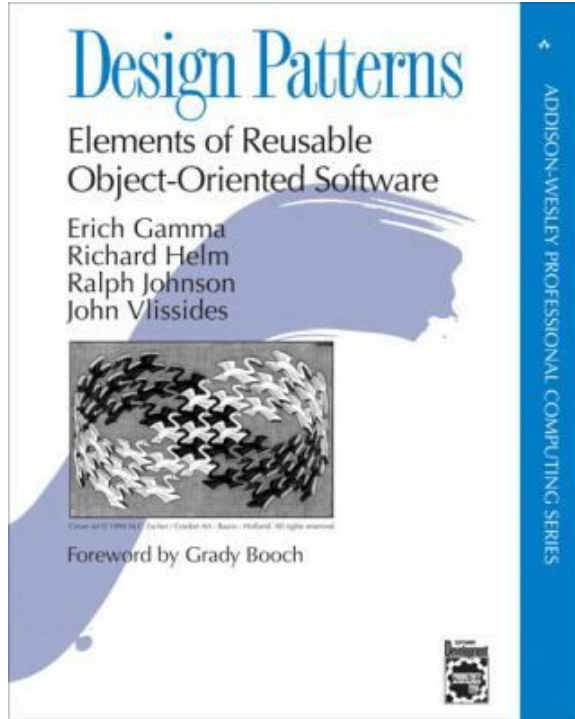
JPL

Jet Propulsion Laboratory



Apply for Internships!

Course Text



Design Patterns by Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides

Not required, but is the origin for most of the slides and makes a great desk reference

It has also been **highly** recommended by students from previous quarters

A few copies are also available to checkout in the library for up to 3 days

GitHub

You will need a GitHub account for this course, starting in your lab **next week**.

It **does not** need to be associated with your UCR NetID.



Review Material

[Basic Linux Navigation](#)

[Basics of g++ Compilation](#)

Inheritance & Polymorphism*

Introduction to the Standard Template Library*

*Review slides on these topics have been posted to iLearn

General Course Procedure

1st Week: Development Methods, Testing

2nd - 8th Week:

- [Lecture] Overview of design pattern (Book)
- [Lecture] Design pattern usage in developing a text editor (Book)
- [Lecture] “Real-world” design pattern usage (Interactive)
- [Lab] Small hands-on development lab (Interactive)

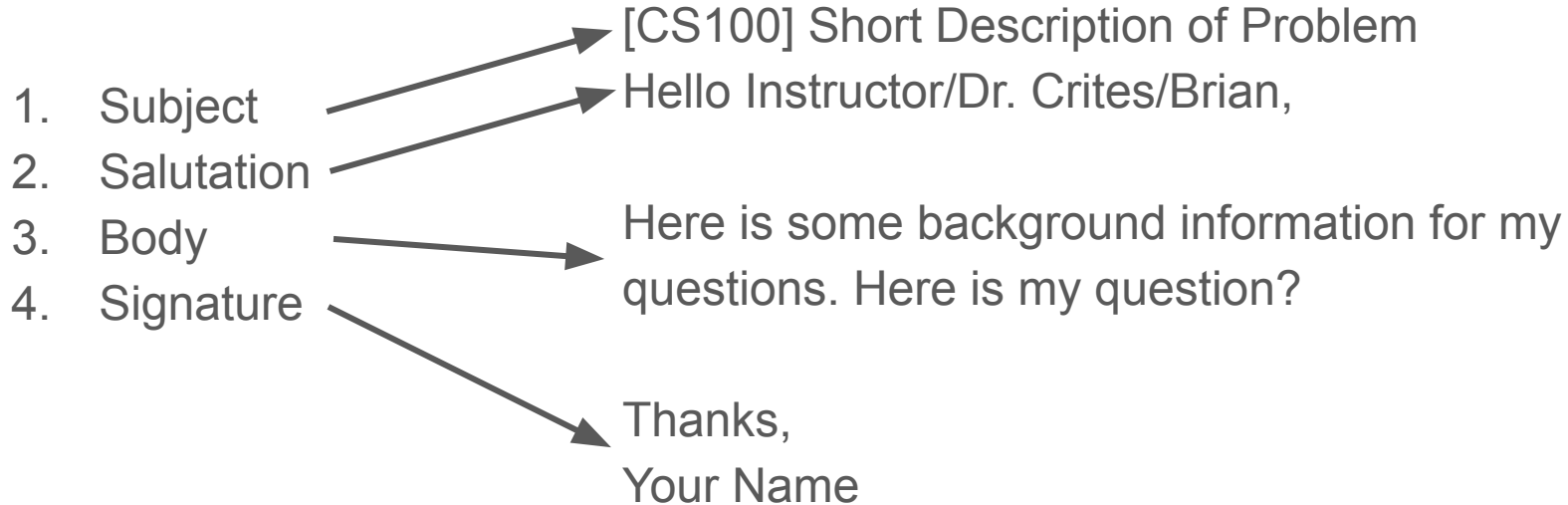
9th Week: Contemporary Issues & Ethics, Developing User Stories

10th Week: Review, Exam (Tentative)

A Note about Course Assignments

- Start looking for a group now (does not need to be someone in your lab)
- Start the assignments when they are assigned (Not starting early enough is one of the top issues students cite every year)
- The assignments are not an application of every design pattern
- Designed to be a “large” scale project for you to continually develop throughout the course
 - Importance of planning before developing
 - Importance of creating extensible, readable, well-designed code
 - Ability to read documentation and use it to develop software

Email



If you go by a different name than appears on iLearn, make sure to include your registered name in any emails about grades

Who Should you Email?

- **Instructor** - Questions about lecture or lecture material (agile methods, testing methods, design patterns, etc.), general queries about exam questions
- **Teaching Assistant** - Questions about lab grades, requesting a lab absence, or to transfer lab sections
- **Course Grader** - Questions and disputes about assignment grades or questions about assignment testing procedures
- **Gradescope Regrade Request** - Exam grade disputes

If you need to be added to iLearn because you are not officially enrolled in the course, please email (bcrit001@ucr.edu) with your request

Cheating & Self Plagiarism

- We know solutions to the assignment exist online, referencing or copying some or all of someone else's code is **cheating**
- Some of you may be repeating this course but you **must email the instructor for approval** to reuse code from a previous quarter
- Cheating or failing to gain prior approval for code reuse will result in an F in the course and follow up with judicial review, **NO EXCEPTIONS!**

Questions?