EECS Course Enrollment Announcement

- Between now and noon on January 7, you can tell students to submit a waiting list request online via the link on the departmental web page.

- After January 7, please refer the student to the EECS undergraduate office in LAS 1012M. We will give the student permission to enroll if he or she has the prerequisites and space permits. (Starting on January 17, students will also be required to get the instructor's signature on a form provided by the undergraduate office in order to enroll.)

EECS 4313

Software Engineering Testing

Topic 00:

Introduction and Admin

Zhen Ming (Jack) Jiang

Course Information

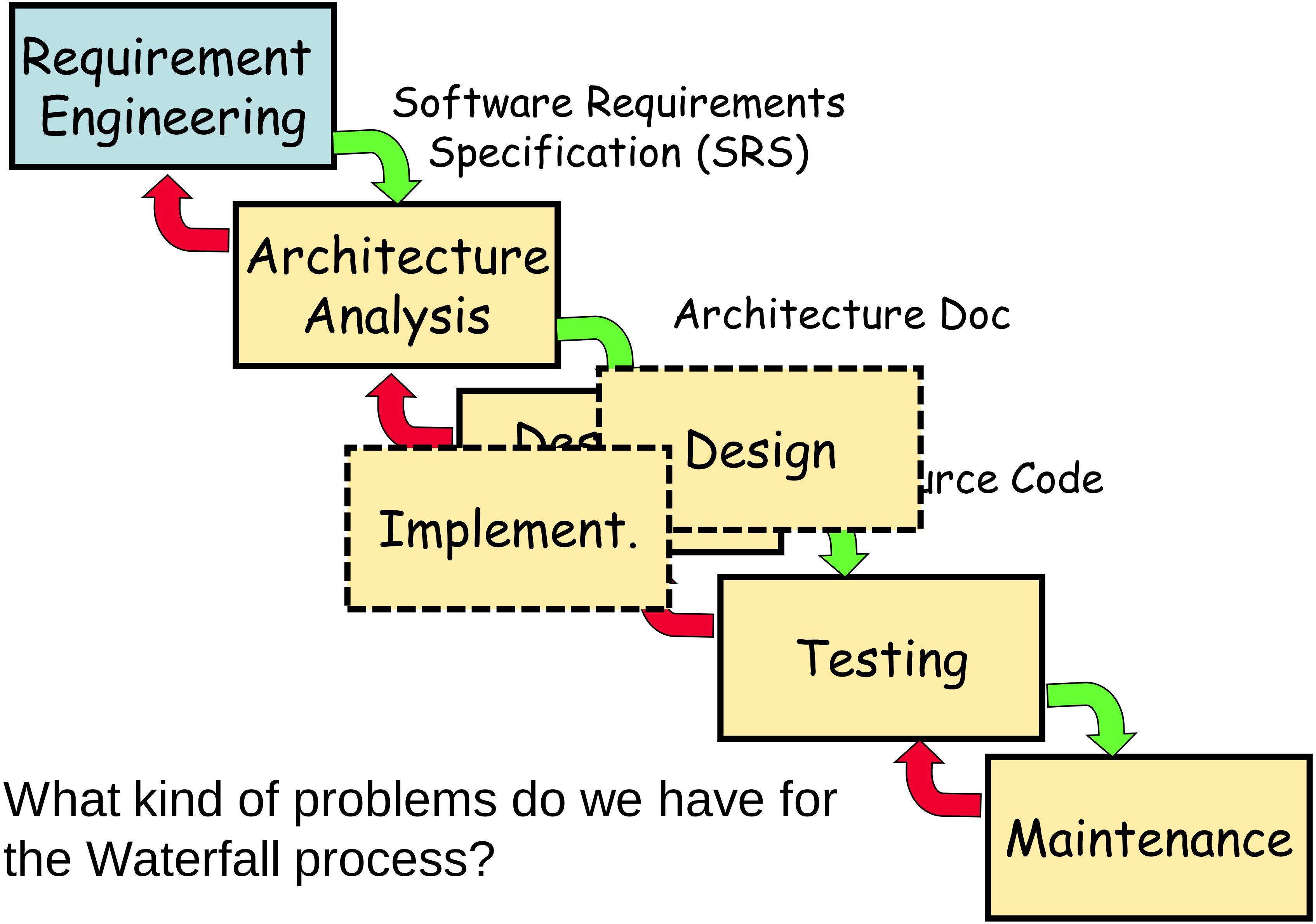
- Lecture Time
  - 2:30 - 4:00 pm
    - every Mondays and Wednesdays at DB 1005
- Instructor:
  - Dr. Zhen Ming (Jack) Jiang, LAS 1012E, zmjiang@cse.yorku.ca
  - Office Hours: Mondays 4:00 - 5:00 pm or by appointment
- Course TA:
  - Mr. Minke Xiu
- Course Webpage:
  - https://wiki.eecs.yorku.ca/course_archive/2018-19/W/4313/
- Send emails from your York's email account, otherwise likely to be flagged as spam
- Put "**EECS4313**" in subject to go around spam filters

On Software Engineering

- Software engineering is a pure intellectual activity
  - Output is documentation
  - Program text is a form of electronic documentation

- Difference with other engineering disciplines
  - Software has no physical characteristic
    - no mass, no heat produced
  - Software implements highly complex functions in a flexible way, making it an essential part of other systems

Waterfall Development Process



What kind of problems do we have for the Waterfall process?

What is Course is about

- This course gives an introduction to systematic methods of testing and verification, covering a range of static and dynamic techniques and their use within the development process. The course emphasizes the view that design should be carried out with verification in mind to achieve overall project goals.

Learning Objectives

- Learn testing techniques and the situations in which they apply
- Apply real-world testing tools and frameworks
- Learn how to file bug reports
- Understand and apply different manual and automated software testing techniques
- Understand the importance of systematic testing

Topics covered

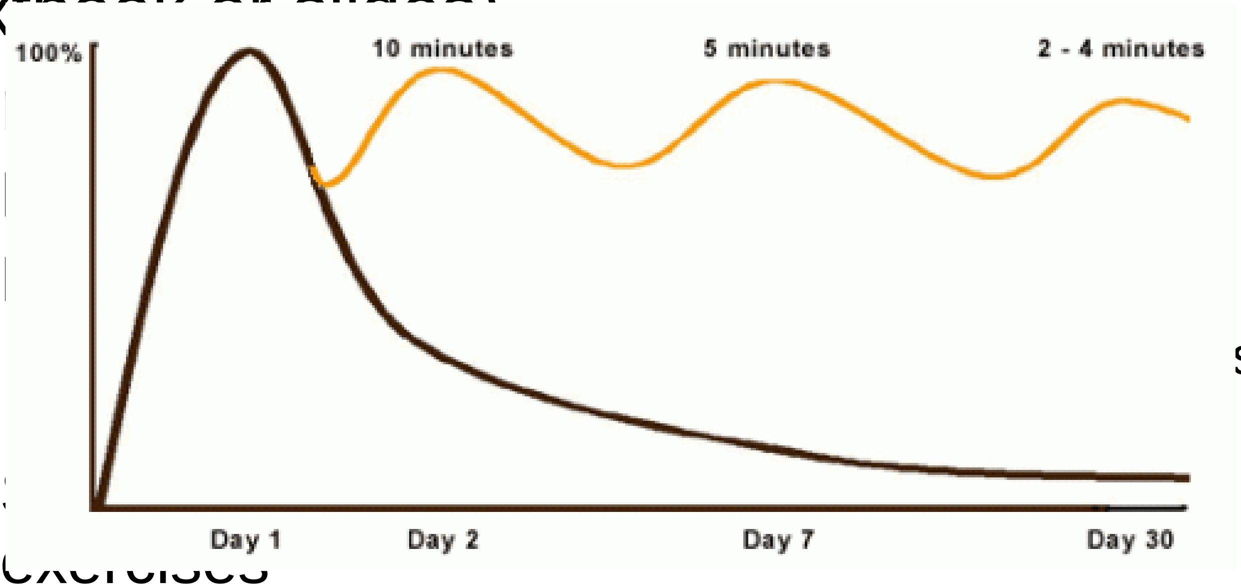
- Limits and objectives of software testing
- Reporting and analyzing bugs
- Black-box testing techniques
- White-box testing techniques
- Non-functional testing techniques (if time permits)

Course Expectations

- Read assigned readings
- Attend lectures and participate in discussions
- Bring your ideas and concerns to class
- Work effectively in a group setting (group members will evaluate each other)
- Learn how to use the tools and understand your project *very well*
- Hand in your deliverables on time

Study Strategy

- Don’t fall behind – Learning is work and repetition
- Attend classes (even though some materials are in textbook or slides)
- Fo
  -
  -
    signments,
  - the
    exercises
    - If you do not reflect on and use the material the same day you forget 50% within 24 hours and 80% within 48 hours



Evaluation

Assignments	35% (3)
Midterm	20%
Final Exam	45%

Working in Groups and Choosing a Group

- Group Size
- Understand the work habits and goals of your group members:
    - Night person
    - Start early
    - Laid back
    - Best project ever
    - Morning person
    - Start at last minute
    - Perfectionist
    - Reasonable mark
- Identify members with good communication skills

Peer Reviews

- All members should receive same marks for project. However, to account for that individual effort, we have peer reviews
  - You can assign each member (including yourself) a grade
  - You have 5 * N + 1 marks, where N is size of group
- Peer reviews are sent 24 hours ***after*** each large deliverable:
  - For example, if assignment 1 is due on 9 am Jan. 21, 2017, the peer review will be due 9 am Jan. 22, 2017.
- **Reviews are submitted via email with subject: "Peer Review for Group ##"**

Lateness Policy for All Course Deliverables

- For all deliverables:
  - Hand hard copy (if specified) at the beginning of class or earlier to the instructor
  - Submit online via PRISM

NO LATE DELIVERABLES!!

Academic Integrity and Cheating

- Cheating, plagiarism and other forms of academic fraud are taken very seriously by the University, the Faculty, and the teaching staff.
- Examples:
  - Submitting the work of another person as your original work
  - Incorporating others work in your work and not referencing it
  - It is permitted and encouraged to discuss assignments with your peers on the whiteboard but **NOT** permitted to copy their solutions as they talk to you. Both parties will be penalized

Course Text

- There are two optional textbooks for the course:
  - Paul C. Jorgensen. Software Testing: A Craftsman's Approach, CRC Press, 2002.
    - *Called the “Jorgensen” book*
  - Paul Ammann and Jeff Offutt. Introduction to Software Testing. Cambridge University Press, Cambridge, UK, 2016. (2nd edition)
    - *Called the “Ammann and Offutt” book*
- Lecture slides, papers, online books
- Additional online readings assigned for case studies
- The exams will cover assigned readings and topics covered in class

Asking Questions

- Ask in class
- Ask me (email, office hours)
- Discuss with your classmates or group members
- Ask on the discussion forum

Tell Me a Little About You

- Background
  - SE, ECE, CS, CE, others?
- Any industrial experience (e.g., co-op or internship)?
- Schedule
  - Any classes after this?
- Plan to graduate in the summer?
    - Grad schools or jobs?

Acknowledgment

- Some of the covered material are based on the materials of Prof. Alex Orso, Bil Tzerpose, Gunnar Gotshalks and Jeff Offutt