CS4004
Software Testing & Inspection
Lecture 1 - Overview

Dr Faeq Alrimawi
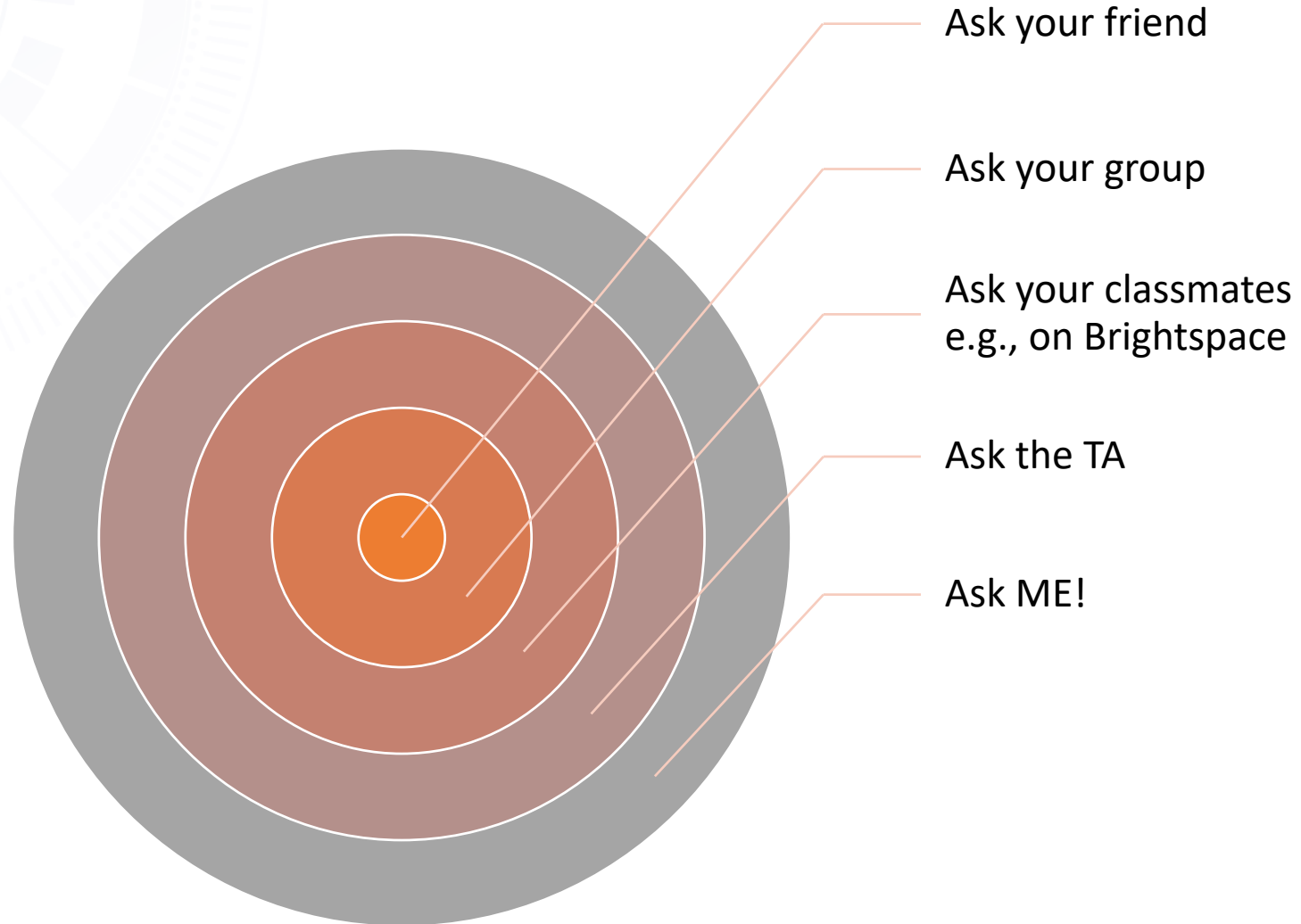
The Team

- Faeq Alrimawi (Facilitator/Instructor)
  - Assistant Professor, CSIS, UL
  - PhD Software Engineering, UL
  - Postdoc in Software Engineering, Lero, UL

- TA: Matthew O'Dwyer

My Escalation Policy

Ask your friend

Ask your group

Ask your classmates e.g., on Brightspace

Ask the TA

Ask ME!

My Escalation Policy - Exception

Ask a friend

Ask your group

Email Me Directly for Personal Issues

Faeq.Alrimawi@ul.ie

If you email me...

Please use **subject line**
[CS4004]

Module Overview

Rationale and Purpose of the Module

Syllabus

Learning Outcomes

Lectures & Labs

Assessment

Forum and E-tivity Participation

Access to the Module Content

Rationale

- This module introduce software **testing** and **inspection**

- **Testing** is an investigation
  - To provide information about the **quality** of the **software** product or service under test

- **Inspection** in software engineering, is
  - **Peer review** of the software product by **trained individuals** who look for defects of software

- **Software testing is the process of evaluating a system with the intent of finding bugs**
  - It is performed to check if the system **satisfies** its specified **requirements**.

Syllabus (1-2)

- Key Terminology
  - testing, debugging, error, bug, defect, ...

- Test types and their place in the software development process
  - Black-box and white-box testing
  - Program reading and comprehension
  - Refactoring code

- Inspections, walkthroughs and desk-checking

Syllabus (2-2)

- Programming with **assertions**

- Using a **debugger** for white-box testing

- **Reporting** and analysing bugs: content of the problem report, analysis of a reproducible bug, making a bug reproducible

- **Test case design**: characteristics of a good test, equivalence classes and boundary values

- **Test case execution** and regression testing

- **Requirements** for white-box and black-box testing tools

Learning Outcomes

- **Explain** the key **terminology** in software testing and inspection.

- **Understand** various **testing techniques** such as black and white box.

- **Write** the tests, **run** them, and **report** the errors.

Bibliography

- **Software Testing** (second edition)
  Paul Ammann and Jeff Offut

- **Software Testing and Analysis: Process, Principles and Techniques**
  Mauro Pezze and Michal Young

- **Software Engineering** (tenth edition)
  Ian Sommerville
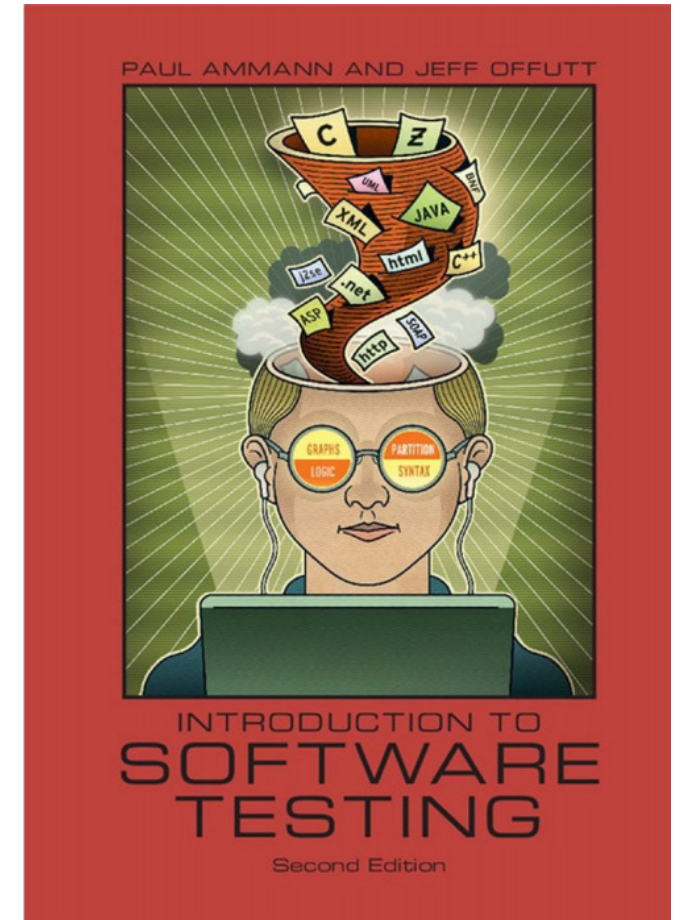
Lectures and Labs (1-3)

I use a **combined teaching** methods

- Classroom **lectures** and **labs**,
  - Project-oriented
  - Problem-oriented

- It allow students to apply these concepts

- They choose and **adapt technologies** to develop solutions for given **problems** and realistic **scenarios** (Creative) in groups (Collaborative).
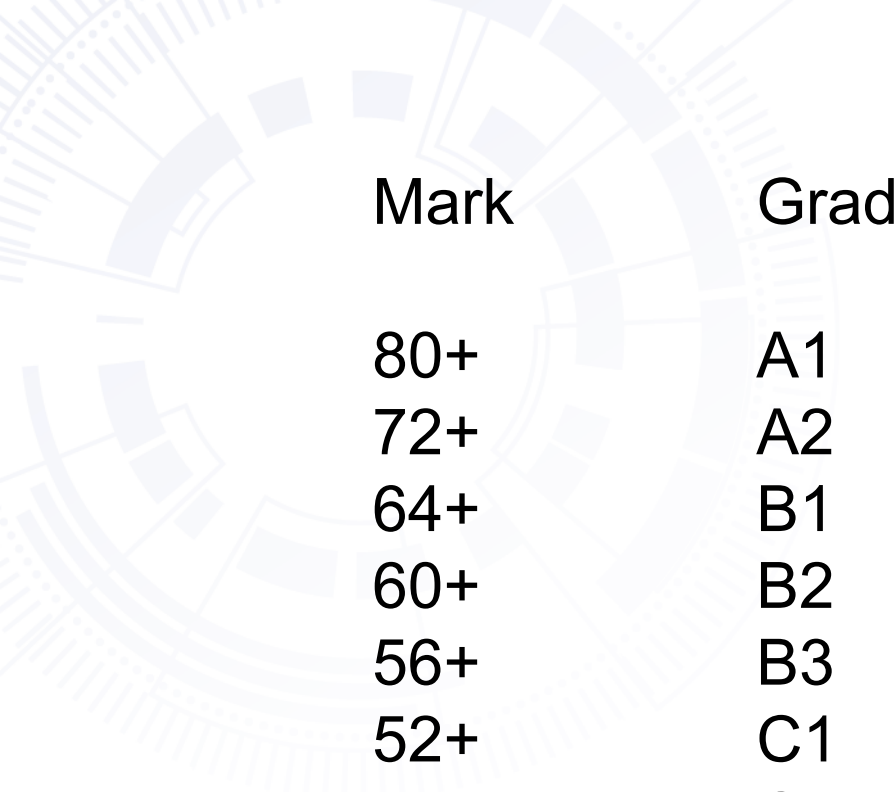
Lectures and Labs (2-3)

- In many cases, the addressed problems and scenarios will include aspects like
  - social challenges,
  - local and global impact,
  - ethical considerations,
  - contributions to society,
  - the active use of technology and research to drive positive change.

- Students will have to **argue** their **rationale**, e.g., for technical choices and design decisions and **present the results** of their work.

Lectures and Labs (3-3)

Week	Lecture	Lab
W1	Module Overview	None
W2	Why we Test? & Terminology	Introduction to the LAB required tools
W3	Bug Reporting	Remote Repository (Jira 1)
W4	MDTD and Criteria Based	Jira II
W5	Criteria-Based Testing	Jira III
W6	Functional Testing and Combinatorial Testing	Reserved
W7	Structural Testing	JUnit Testing Exercise
W8	Test Driven Development	JUnit II
W9	Data Flow Testing	JUnit III
W10	Inspection	JUnit IV
W11	Module recap and info about assignment	Reserved
W12	Reserved	Reserved

Module Assessment

Due	Percentage	Assessment Type
Week 5	15%	Group Assignment
Week 8	25%	Midterm Exam
Week 10	20%	Group Assignment
Week 13	40%	Individual Final Project

Mark	Grade	QCA
80+	A1	4.00
72+	A2	3.60
64+	B1	3.20
60+	B2	3.00
56+	B3	2.80
52+	C1	2.60
48+	C2	2.40
40+	C3	2.00
35+	D1	1.60
30+	D2	1.20
0	F	0.00
	NG	0.00
	I	incomplete

Grading Scheme

Brightspace Participation (1-2)

- Forums are an important part of your learning journey.
  - Allow you to **socialise** with your **peers**
  - Help create a **collaborative community** where you can ask and answer questions.
  - Forums also give you the opportunity to clarify general points **with your module leader/ TA** in an open environment.
  - By asking these questions on a forum, rather than in an email, you **allow your peers to also benefit** from the answers.

- We want to keep **the quality of posts high** so that the forums remain a valuable learning resource
  - So please read the forum guidelines listed below.
  - Netiquette Tips For Online Discussions.

Brightspace Participation (2-2)

- Post questions to the forums first and wait for the answer of your peers, or module leader/TA
  - Only personal issues should be sent to the module leader via e-mail
  - Make sure that other students can see both, the question and the responses and to allow for efficient communication
- Do not post about the same issue in multiple locations
- Engage on forums regularly
- Engage in E-tivities (e.g. quizzes)

Peer-Based Learning

- **Contribute** to peer-based learning
- **Engage** in group work regularly
- For brief issues, you should aim to **respond** to your group members on the same day (during the week), but **no later than 24 hours**.
- It is also a good idea to agree on how and how often you will **communicate** within your group at the start of the group work
- You need to agree on a **communication tools**

Access to the Module Content

Open the module (CS4004) in Brightspace

- Go over every week's module content in its respective weeks
- Engage with other students on module content, for example on the forums
- Ask for help early
- If you struggle, please get in touch immediately rather than disappearing and struggling by yourself

Remember, I am always ready to help you and I want you to do well!

Additional Notes

- **Contribute** in **Brightspace systematically** every week

- **Attend** the **class** as much as possible

- **4 per group** is the best
  - Start thinking about creating a project group.

What Are Your Questions?