Validation and Verification (COMP0103)

Description

Aims:
The module will train students in the principles and techniques of validating and verifying software systems. The training will be intellectually demanding and will cover not only the state-of-the practice in validation and verification, but also the most significant trends, problems and results in validation and verification research.

Learning outcomes:
On successful completion of the module, a student will be able to:
1. understand correctness, consistency, faults and failures, static analysis and testing;
2. understand the use of logic as a formal language for the specification of systems;
3. understand the use of symbolic execution;
4. verify simple systems;
5. understand the range of approaches to testing that can be applied to software systems;
6. undertake both black-box and white-box (unit-level) testing;
7. appreciate the limitations of the current tools and have insights in ongoing research topics to overcome them;

Content:
Basic concepts:
- Software engineering lifecycle context;
- Correctness;
- Soundness and completeness;
- Faults;
- Errors;
- Failures;
- Static and dynamic analysis;

Validation:
- Kinds of testing (unit, functional, integration, system, acceptance, regression);
- Black box and White box testing;
- Input partitioning and Random Testing;

Key information

Year: 2019/20
Credit value: 15 (150 study hours)
Delivery: PGT L7, Campus-based
Reading List: View on UCL website
Tutor: Dr Federica Sarro
Term: Term 2
Timetable: View on UCL website

Assessment

- Written examination (main exam period): 80%
- Coursework: 10%
- Coursework: 10%

Find out more

For more information about the department, programmes, relevant open days and to browse other modules, visit ucl.ac.uk

Disclaimer: All information correct as of June 2019. Please note that aspects of the module may be subject to change. UCL will make best efforts to inform applicants of major changes.
-Coverage and Structural Testing;
-Mutation Testing;
-Regression Testing;

Verification:
-Propositional and Predicate Logic;
-Specifying and verifying programs;
-Symbolic Execution;
-Hoare Logic;

Reading: Selected surveys and research papers;

Requisites:
In order to be eligible to select this module, a student must be registered on a programme for which it is a formally-approved option or elective choice AND must have a background equivalent to Years 1 and 2 of BSc/ MEng Computer Science UCL.
Validation and Verification (COMP0103)

Description

Aims:
The module will train students in the principles and techniques of validating and verifying software systems. The training will be intellectually demanding and will cover not only the state-of-the practice in validation and verification, but also the most significant trends, problems and results in validation and verification research.

Learning outcomes:
On successful completion of the module, a student will be able to:
1. understand correctness, consistency, faults and failures, static analysis and testing;
2. understand the use of logic as a formal language for the specification of systems;
3. understand the use of symbolic execution;
4. verify simple systems;
5. understand the range of approaches to testing that can be applied to software systems;
6. undertake both black-box and white-box (unit-level) testing;
7. appreciate the limitations of the current tools and have insights in ongoing research topics to overcome them;

Content:
Basic concepts:
- Software engineering lifecycle context;
- Correctness;
- Soundness and completeness;
- Faults;
- Errors;
- Failures;
- Static and dynamic analysis;

Validation:
- Kinds of testing (unit, functional, integration, system, acceptance, regression);
- Black box and White box testing;
- Input partitioning and Random Testing;

Key information

Year 2019/20
Credit value 15 (150 study hours)
Delivery UGM L7, Campus-based
Reading List View on UCL website
Tutor Dr Federica Sarro
Term Term 2
Timetable View on UCL website

Assessment

- Written examination (main exam period): 80%
- Coursework: 10%
- Coursework: 10%

Find out more

For more information about the department, programmes, relevant open days and to browse other modules, visit ucl.ac.uk

Disclaimer: All information correct as of June 2019. Please note that aspects of the module may be subject to change. UCL will make best efforts to inform applicants of major changes.
- Coverage and Structural Testing;
- Mutation Testing;
- Regression Testing;

Verification:
- Propositional and Predicate Logic;
- Specifying and verifying programs;
- Symbolic Execution;
- Hoare Logic;

Reading: Selected surveys and research papers;

**Requisites:**
In order to be eligible to select this module, a student must be registered on a programme for which it is a formally-approved option or elective choice AND must have a background equivalent to Years 1 and 2 of BSc/ MEng Computer Science UCL.