

Academic Org: Div of Computer Science & Engg – Subject: Computer Science

Course: CSCI5700 **Course ID:** 015391 **Eff Date:** 2026-07-01 **Crse Status:** Active **Apprv. Status:** Approved **[New Course]**
Advanced Topics in Software Testing 進階軟件測試

This course will focus on practical software engineering topics relevant to software testing. We will explore advanced modern software testing techniques to detect security vulnerabilities, functional defects, and more. This course will explain in detail the most advanced topics, including automated test generation, fuzzing, symbolic execution, and AI-powered test generation. We will also show the state-of-the-art software testing solutions based on generative AI models.

本科將專注於與軟件測試相關的實用軟件工程議題。本科將探討先進的現代軟件測試技術，用以檢測安全漏洞、功能缺陷等問題。本科將深入講解最前沿的主題，包括自動化測試生成、模糊測試（fuzzing）、符號執行以及人工智能驅動的測試生成。同時亦會展示基於生成式人工智能模型的最新軟件測試方案。

Grade Descriptor: A

EXCELLENT – exceptionally good performance and far exceeding expectation in all or most of the course learning outcomes; demonstration of superior understanding of the subject matter, the ability to analyze problems and apply extensive knowledge, and skillful use of concepts and materials to derive proper solutions.

有關等級說明的資料，請參閱英文版本。

B

GOOD – good performance in all course learning outcomes and exceeding expectation in some of them; demonstration of good understanding of the subject matter and the ability to use proper concepts and materials to solve most of the problems encountered.

有關等級說明的資料，請參閱英文版本。

C

FAIR – adequate performance and meeting expectation in all course learning outcomes; demonstration of adequate understanding of the subject matter and the ability to solve simple problems.

有關等級說明的資料，請參閱英文版本。

D

MARGINAL – performance barely meets the expectation in the essential course learning outcomes; demonstration of partial understanding of the subject matter and the ability to solve simple problems.

有關等級說明的資料，請參閱英文版本。

F

FAILURE – performance does not meet the expectation in the essential course learning outcomes; demonstration of serious deficiencies and the need to retake the course.

有關等級說明的資料，請參閱英文版本。

Equivalent Offering:

Units: 3 (Min) / 3 (Max) / 3 (Acad Progress)

Grading Basis: Graded

Repeat for Credit: N

Multiple Enroll: N

Course Attributes: MSc Computer Science
MPhil-PhD Computer Sci & Erg

Topics:

COURSE OUTCOMES

Learning Outcomes:

At the end of the course of studies, students will be able to:

1. Fundamental understanding of the pros and cons of various dynamic and static software testing tools;
2. Conduct extensive testing on various software systems to ensure their reliability;
3. Conduct research in software engineering and software security-related fields.

Course Syllabus:

Week 1: Software testing and software quality assurance

Week 2: Program analysis basics

Week 3: Random test generation
Week 4: Grammar-based test generation
Week 5: Mutation-based testing
Week 6: Guided random testing
Week 7: Symbolic and concolic testing
Week 8: Regression testing
Week 9: AI-based test generation
Week 10: Testing advanced software: Compilers and databases
Week 11: Testing advanced software: Deep learning libraries
Week 12: Final projects
Week 13: Final projects

Assessment Type:

Homework or assignment	: 30%
Project	: 70%

Feedback for Evaluation:

1. Course evaluation and questionnaire
2. Question-and-answer sessions during class
3. Student consultation during office hours or online

Required Readings:

To be provided by course teacher.

Recommended Readings:

- Cadar, Cristian, Daniel Dunbar, and Dawson R. Engler. "Klee: unassisted and automatic generation of high-coverage tests for complex systems programs." OSDI.. Vol. 8. 2008.
- Majumdar, Rupak, and Koushik Sen. "Hybrid concolic testing." 29th International Conference on Software Engineering (ICSE'07). IEEE, 2007.
- Holler, Christian, Kim Herzig, and Andreas Zeller. "Fuzzing with code fragments." 21st USENIX Security Symposium (USENIX Security 12). 2012.

- Le, Vu, Mehrdad Afshari, and Zhendong Su. "Compiler validation via equivalence modulo inputs." ACM Sigplan Notices 49.6 (2014): 216-226.

- Ba, Jinsheng, and Manuel Rigger. "Testing database engines via query plan guidance." 2023 IEEE/ACM 45th International Conference on Software Engineering (ICSE). IEEE, 2023.

- Ernst, Michael D., et al. "Dynamically discovering likely program invariants to support program evolution." IEEE transactions on software engineering 27.2 (2001): 99-123.

- Zeller, Andreas, and Ralf Hildebrandt. "Simplifying and isolating failure-inducing input." IEEE Transactions on software engineering 28.2 (2002): 183-200.

OFFERINGS

1. CSCI5700 Acad Organization=CSEGV; Acad Career=RPG

COMPONENTS

LEC : Size=30; Final Exam=N; Contact=3
TUT : Size=30; Final Exam=N; Contact=1

ENROLMENT REQUIREMENTS

1. CSCI5700

Enrollment Requirement Group:

For students in MSc Computer Science; or
For students in MPhil-PhD Computer Science & Engineering; or
For students in UG AISTN or CDASN or CENGN or CSCIN

New Enrollment Requirement(s):

Other Requirement = For students in MSc Computer Science; or
For students in MPhil-PhD Computer Science & Engineering; or
For students in UG AISTN or CDASN or CENGN or CSCIN

Additional Information

VTL-Onsite face-to-face hrs 0
VTL-Online synch. hrs 0
VTL-Online asynch. hrs 0

< E N D O F R E P O R T >