1. **Course Number and Course Title:**

   CMP 494-11 – Software Testing

2. **Credit Hours:**

   3 – 0 – 3

3. **Prerequisites and/or Co-Requisites:**

   Prerequisite: CMP 305 (Data Structures and Algorithms)

4. **Name and Contact Information of Instructor:**

   Name: Dr. Khaled El-Fakih
   Email: kelfakih@aus.edu
   Office: ESB 2065
   Phone: 06 515 2492
   Office Hours: Posted on office door and iLearn, also by appointment.

5. **Course Description (Catalog Description):**

   Provides an overview of software testing and applicable techniques. Covers black-box and white-box testing, code-based and model-based mutation testing, conformance testing, and regression testing. Introduces the application of reinforcement learning, genetic algorithms, and simulated annealing in software testing.

6. **Textbook and other Supplemental Material:**

   Textbook:
   - A. Mathur, *Foundations of Software Testing*, 2nd ed., Pearson, 2013.

   Other supplemental material:
   - Z. Kohavi, N. Jha, *Switching and Finite Automata Theory*, 3rd ed., Cambridge University Press, 2009.
   - P. Jorgensen, *Software Testing: A Craftsman's Approach*, 5th ed., Taylor & Francis, 2021.
   - T. King, *AI-Driven Testing,* O'Reilly, 2021.
   - R. Sutton, A. Barto, *Reinforcement Learning: An Introduction*, 2nd ed., Bradford Books, 2018.

7. **Course Learning Outcomes:**

   Upon completion of the course, students will be able to:
   1. Describe software testing processes and activities.
   2. Generate tests using mutation testing.
   3. Derive conformance test cases for specifications modeled as finite state machines.
   4. Apply reinforcement learning, genetic algorithms, and simulated annealing in testing.
   5. Construct functional tests using black-box boundary value and equivalence class testing, decision tables and cause effect graphs.
   6. Build white-box dataflow and structural tests (code-based testing).
   7. Demonstrate a deep understanding of regression testing methods.
   8. Develop appropriate test derivation methods for selected applications.

8. **Teaching and Learning Methodologies:**

   Methods include lectures, class discussions, homework and reading assignments, and a course project.

9. Course Topics and Schedule:

Topic/Activity	Weeks
Overview of software testing methods and tools	Week #1
Model-based mutation testing	Week #2
Code-based mutation testing	Week #3
Distinguishing sequences in conformance testing	Week #4
Test derivation in conformance testing	Week #5
Introduction to reinforcement learning	Week #6
AI in testing: Reinforcement learning in testing; Midterm 1	Week #7
AI in testing: Genetic and simulated annealing algorithms in testing	Week #8
Boundary value and equivalence class testing (black box)	Week #9
Decision table and cause effect graph testing	Week #10
Code-based testing (white box); Midterm 2	Week #11
Code based structural testing	Week #12
Regression testing	Week #13
Project presentations	Week #14
Project demos; Revision	Week #15
Final exam	Week #16

10. Schedule of Laboratory and other Non-Lecture Sessions:

Labs: This course has no labs.

Project: Groups of 3 to 4 students will work on the design, implementation, and assessment of selected test derivation methods. The project deliverables will be a report, a presentation, and the source code of the implementation.

11. Out-of-Class Assignments with Due Dates:

Assignment	Due Date (tentative)
Homework 1 – Mutation testing	Week #3
Homework 2 – Distinguishing sequences	Week #5
Homework 3 – Conformance testing	Week #7

12. Student Evaluation:

Assessment	Weight	Due Date (tentative)
Homework assignments	5 %	cf. section 11
Midterm 1	16 %	Week #7
Midterm 2	16 %	Week #11
Project	28 %	Week #14
Final exam	35 %	Week #16

13. Assessment Instruments:

Assessment	Course Learning Outcomes
Homework assignments	O2, O3, O5–O7
Midterm 1	O1–O4
Midterm 2	O5–O7
Project	O1–O8
Final exam	O1–O7

14. Contribution of Course to Program Outcomes:

BSCS Program Outcomes	Emphasis in this course	Course Learning Outcomes
(1) Analyze a complex computing problem and apply principles of computing and other relevant disciplines to identify solutions.	◑	O1–O7
(2) Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline.	◑	O2–O6, O8
(3) Communicate effectively in a variety of professional contexts.	○	O8
(4) Recognize professional responsibilities and make informed judgments in computing practice based on legal and ethical principles.		
(5) Function effectively as a member or leader of a team engaged in activities appropriate to the program's discipline.	○	O8
(6) Apply computer science theory and software development fundamentals to produce computing-based solutions.	◑	O2–O8

Emphasis: ● High; ◑ Medium; ○ Low; Blank – Nothing Specific Expected

BSCoE Program Outcomes	Emphasis in this course	Course Learning Outcomes
(1) Identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics	◑	O1–O7
(2) Apply engineering design to produce solutions that meet specified needs with consideration of public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors		
(3) Communicate effectively with a range of audiences	○	O8
(4) Recognize ethical and professional responsibilities in engineering situations and make informed judgments, which must consider the impact of engineering solutions in global, economic, environmental, and societal contexts		
(5) Function effectively on a team whose members together provide leadership, create a collaborative and inclusive environment, establish goals, plan tasks, and meet objectives	○	O8
(6) Develop and conduct appropriate experimentation, analyze and interpret data, and use engineering judgment to draw conclusions	◑	O2–O6, O8
(7) Acquire and apply new knowledge as needed, using appropriate learning strategies	○	O2–O8

Emphasis: ● High; ◑ Medium; ○ Low; Blank – Nothing Specific Expected

15. Letter Grade Policy:

Total (T)	Letter Grade
$94 \leq T$	A

$90 \leq T < 94$	A-
$85 \leq T < 90$	B+
$80 \leq T < 85$	B
$75 \leq T < 80$	B-
$70 \leq T < 75$	C+
$65 \leq T < 70$	C
$60 \leq T < 65$	C-
$55 \leq T < 60$	D
$T < 55$	F