1. **Course number and name**
   CMP 454 - Software Testing and Quality Engineering

2. **Credits and contact hours**
   3 credit hours, 3 contact hours

3. **Instructor's or course coordinator's name**
   Dr. Khaled El-Fakih

4. **Textbook, title, author, and year**
   A. Mathur, *Foundations of Software Testing*, Addison-Wesley, 2008.

   Other supplemental materials
   P. Jorgensen, *Software Testing: A Craftsman's Approach*, 2nd edition, CRC Press, 2002.
   Z. Kohavi, *Switching and Finite Automata Theory*, McGraw-Hill Book Company, NY, 1978.
   Slides at *ilearn* folder

5. **Specific course information**
   a. **Brief description of content of the course (catalog description)**
      Provides an overview of software engineering. Covers software quality assurance; black-box and white-box testing; integration and regression testing; and selected topics from the following: object-oriented software testing, acceptance testing, conformance testing, diagnostic testing, test execution, distributed systems testing, test languages and test tools, GUI testing, interoperability testing, test metrics, and standards for software quality and testing.

   b. **Prerequisites or co-requisites**
      Prerequisites: CMP 305/COE 311 (Data Structures and Algorithms)
      Prerequisite/Concurrent: CMP 350/COE 420 (Software Engineering)

   c. **Indicate whether a required, elective, or selected elective course in the program**
      Selected Elective

6. **Specific goals for the course**
   a. **Specific outcomes of instruction**
      Upon completion of the course, students will be able to:
      1. Describe software testing processes and activities
      2. Construct functional tests using boundary value and equivalence class testing methods
      3. Develop functional tests using decision tables and cause effect graphs
      4. Build structural code-based tests
      5. Derive test cases for specifications modeled as finite state machines
      6. Describe integration and regression testing methods

b. Explicitly indicate which of the student outcomes listed in Criterion 3 or any other outcomes are addressed by the course

This course contributes in a significant way to the accomplishment of the following program outcomes:

Program outcome	Emphasis in this course
(a) an ability to apply knowledge of computing and mathematics appropriate to the discipline	◑
(b) an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution	○
(c) an ability to design, implement, and evaluate a computer-based system, process, component or program to meet desired needs	
(d) an ability to function effectively on teams to accomplish a common goal	○
(e) an understanding of professional, ethical, legal, security and social issues and responsibilities	
(f) an ability to communicate effectively with a range of audiences	○
(g) an ability to analyze the local and global impact of computing on individuals, organizations, and society	
(h) recognition of the need for and an ability to engage in continuing professional development	○
(i) an ability to use current techniques, skills, and tools necessary for computing practice	◑
(j) an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices	◑
(k) an ability to apply design and development principles in the construction of software systems of varying complexity	

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

7. Brief list of topics to be covered

i. Boundary value and Equivalence testing
ii. Decision table (and Cause effect) testing
iii. Dataflow and Structural
iv. Integration and Regression testing
v. Conformance/Protocol testing
vi. Mutation Testing