CMPE110-Fundamentals of Computing and Programming

Department: Computer Engineering

Instructor information
Prof. Dr. Omar Ramadan, **E-mail:** omar.ramadan@emu.edu.tr, **Office:** CMPE115, **Tel.:** 1194

Meeting times and places
Tuesday: 10:30-12:20 (CMPE033), Wednesday: 12:30-14:20 (CMPE033), Thursday:16:30-18:20 (CMPE137)

Program Name: Computer Engineering	**Program Code:** 25
Course Number: CMPE 110	**Credits:** 4 Cr	**Year/Semester:** 2019-2020, Fall

☒ Required Course ☐ Elective Course

Prerequisite(s):
None

Catalog description:
This course presents the basic description of computer hardware and software. Also, it introduces the basics of problem solving concept, algorithm, pseudo-code, and flowchart. The fundamentals of computer programming using C++ programming language are also covered.

Course web page:
http://cmpe.emu.edu.tr/courses/cmpe110

Textbook(s): J. Deitel and H. Deitel "C++ How to Program", Global Edition, 10/E, ISBN-10: 1292153342 • ISBN-13: 9781292153346, ©2017 • Pearson

Indicative basic reading list :
1. Walter Savitch, Problem Solving with C++, 9/E, Pearson, 2015 (SBN-13: 978-0133862218)
2. Y. Daniel Liang, Introduction to programming with C++, 3rd edition, Prentice Hall, 2014.
3. M. Sprankle, Problem Solving and Programming Concepts, 8th edition, Pearson Education, 2008.

Topics covered and class schedule (tentative):
(4 hours of lectures per week)

Week 1	Introduction to computer hardware and software. Introduction to computer programming languages.
Week 2	General problem solving concept. Algorithm, pseudo code, and flowchart concept.
Week 3	Introduction to C++ programming language, simple sequential structure.
Week 4, 5	C++ Programming: selective structure.
Week 6, 7	C++ Programming: repetitive structure.
Week 8, 9	Midterm
Week 10,	C++ Programming: functions
Week 11, 12	C++ Programming: one-dimensional and two-dimensional arrays.
Week 13, 14	C++ Programming: introduction to class concept.

Laboratory experiments (tentative):

1.	Introduction and debug features of MS visual C++ compiler
2.	C++ Programming: Sequential logic structure
3.	C++ Programming: selective logic structure
4.	C++ Programming: repetitive logic structure
5.	C++ Programming: Functions
6.	C++ Programming: One and two dimensional arrays
7.	C++ Programming: classes
8.	C++ Programming: Review

Course learning outcomes:
Upon successful completion of the course, students are expected to have the following competencies:
1) Identify the difference between computer hardware and computer software.
2) Construct an algorithm and /or flowchart for solving a problem.
3) Use IDE to edit, compile, and executing C++ code.
4) Understand the basics of C++ high level languages (HLL).
5) Use if-statement and switch statement to implement selective structure program in C++ HLL.

6) Use while-loop, do-while loop, and for-loop to construct repetitive structures in C++ HLL.
7) Ability to use modular programming for implementing multi-task problems in C++ HLL.
8) Ability to use arrays concept in C++ HLL.
9) Ability to use classes concept in C++ HLL (introduction).

	Method	No	Percentage
Assessment <u>**(tentative)**</u>	Quizzes	~2	20%
	Midterm Exam	1	30%
	Final Examination	1	40%
	Lab	~8	10% (5% attendance, 5% performance)

Policy on makeups: For eligibility to take a makeup exam, the student should bring a doctor's report within 3 working days of the missed exam. You will have only one make-up for Midterm or Final exams. Make-up will be organized after final exam period and will cover all material studied during the semester. No make-up will be given to quizzes.

Policy on cheating and plagiarism: Any student caught cheating in the class; exams or assignments will automatically fail the course and may be sent to the disciplinary committee at the discretion of the instructor.

Policy on NG grades: NG grade will be given in case of missing Midterm or Final without official excuse. NG will also be given in case of Lab attendance less than 50%. NG will also be in case of very poor attendance.

Contribution of course to ABET criterion 5
Credit Hours for:
Mathematics & Basic Science : 0
Engineering Sciences and Design : 4
General Education : 0

Relationship of the course to program outcomes
The course has been designed to contribute to the following program outcomes:
1) an ability to identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics.
6) an ability to develop and conduct appropriate experimentation, analyze and interpret data, and use engineering judgment to draw conclusions.

Prepared by: Prof. Dr. Omar Ramadan	**Date Prepared:** Sep. 25th , 2019