**EASTERN MEDITERRANEAN UNIVERSITY  
CMSE 443 Fall 2025 COURSE OUTLINE**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **COURSE CODE** | CMSE 443 | **COURSE LEVEL** | | Fourth year | |
| **COURSE TITLE** | Real-Time System Design | | | | |
| **COURSE TYPE** | Area Elective | | | | |
| **LECTURER(S)** | Assoc. Prof. Alexander Chefranov | | | | |
| **ASSISTANTS** | [HAMIDREZA RAHIMI](https://cmpe.emu.edu.tr/en/about-us/staff/staff-detail?sid=308&n=hamidreza-rahimi" \t "_blank), Mahamat H.M. Hasan | | | | |
| **CREDIT VALUE** | 4 | | **ECTS VALUE** | |  |
| **PREREQUISITES** | CMPE 242 Operating Systems | | | | |
| **COREQUISITES** | None | | | | |
| **DURATION OF COURSE** | 1 Semester | | | | |
| **WEB LINK** | <https://staff.emu.edu.tr/alexanderchefranov/en/teaching/cmpe443/home> | | | | |
| **MEETING TIMES** | Group 1: Monday, 12.30-14.20, CMPE025; Tuesday, 14.30-15.20, CMPE028; Wednesday, 16.30-18.20, CMPE137 (Labs)  Group 2: Tuesday, 8.30: CMPE027; Thursday, 10.30, CMPE027; Frıday, CMPE134 (Labs) | | | | |
| **CATALOGUE DESCRIPTION** Course goal is to introduce students to key ideas, concepts and tools of Real-Time systems design. Introduction to real-time systems, ADA programming, architecture and design of real-time systems, concurrent programming and synchronization, real-time scheduling, reliability and exception handling, real-time OS, and distributed real-time systems | | | | | |
| **AIMS & OBJECTIVES** Course goal is to introduce students to key ideas, concepts and tools of Real-Time systems design. | | | | | |
| **GENERAL LEARNING OUTCOMES (COMPETENCES)** On successful completion of the course, the student is expected to develop **knowledge** and **understanding** of:   * The concept of real-time system * Principles of using of computers to control a plant * Relation between time sampling and rates of the multi-rate system * Features and purposes of hardware used in real-time systems; * Principles of hardware performance increasing * Principles of functioning of sensors used in real-time systems * Principles of real-time systems software organization * Basic principles of real-time operating systems, cooperative and preemptive multitasking * Interruptions and exceptions handling concepts * Notion of shared and critical resources   On successful completion of the course, the student is expected to develop **skills** in:   * Understanding and using notions of a process and thread, their states, contexts, transitions between states * Understanding and using basic scheduling algorithms (Rate-monotonic, Earliest-deadline first), know theoretical results regarding tasks schedulability * Understanding and using synchronization primitives (binary and counting semaphores, mailboxes, events, critical sections, monitors), know relationships between them * Understanding and using deadlock exclusion and avoidance methods (necessary deadlock conditions, Banker’s algorithm) * Understanding and using methods of solving of priority inversion problem (priority inheritance protocols, original and immediate)   On successful completion of the course, the student is expected to develop **abilities** of:   * Understanding and applying methods of designing (top-down decomposition, object-oriented approach, formal logic methods, finite state automata, statecharts, Petri nets, UML diagrams) * Understanding POSIX real-time operating systems requirements * Understanding concepts of task management in ADA * Understanding principles of real-time systems reliability and performance assessment and increasing * Designing of simple real-time systems   On successful completion of this course, the student is expected to develop **appreciation** of   * Principles of Real-Time Systems design * Theoretical achievements in scheduling theory * Practical usage of Real-Times Systems | | | | | |
| **GRADING CRITERIA** Will be decided according to student performance. | | | | | |
|  |  |  |  |  |  |
| **RELATIONSHIP WITH OTHER COURSES** Concepts, methods and tools considered in the course are useful in many branches of computer and software engineering. While the course is self-contained, knowledge and skills gained in mathematical courses, programming languages, hardware courses, low-level programming, and operating systems facilitate the study of this course. The closest courses are CMPE-242 Operating Systems and CMPE-423 Embedded System Design | | | | | |
| **LEARNING / TEACHING METHOD** Mainly through lectures, using Powerpoint slides and the whiteboard, and labs and term project for getting experience. | | | | | |
| **ASSIGNMENTS** There will be 3 labs and one term project. | | | | | |
| **METHOD OF ASSESSMENT**   * 30% Midterm exam * 40% Final exam * 10% Labs * 20% Term Project * 0% Attendance | | | | | |
| **ATTENDANCE** is compulsory. | | | | | |
| **TEXTBOOK** [PHILLIP A. LAPLANTE,SEPPO OVASKA, REAL TIME SYSTEMS DESIGN AND ANALYSIS, 4th EDITION, 2012, ISBN 978-0-470-76864-8](https://staff.emu.edu.tr/alexanderchefranov/Documents/CMSE443/CMSE443%20Spring2020/Laplante2012%20Real-Time%20Systems%20Design%20and%20Analysis.pdf) | | | | | |
| **INDICATIVE BASIC READING LIST** None | | | | | |
| **EXTENDED READING LIST** None | | | | | |
| **SEMESTER OFFERED** 2025-2026 Fall Semester | | | | | |

**CONTENT & SCHEDULE**

* Introduction to RTS
* Hardware considerations. Hardware interfacing, CPU, Memory, I/O, Interruptions, Enhancing performance, Other special devices, non von Neumann architectures
* Real-Time Operating Systems. Real-Time kernels. Theoretical foundations of RTOS. Inter-task communication and synchronization, exceptions. Resource management (processor time, RAM, disk memory). Case study
* Software requirements specification. Formal methods. Finite state machine. Statecharts. Petri nets. Structured analysis and design. Object-oriented analysis and Unified Modeling Language. Software engineering principles.
* Programming languages and software production process. Assembly language. Procedural languages. Object-oriented languages. ADA 95 tasks, rendezvous, pragmas. Special RT languages.
* System performance analysis and optimization. Response time and time loading. Scheduling problems. calculation. Optimization technique. Analysis of memory requirements. Queuing models. Little's Law.

Laboratory Works Tentative Schedule

|  |  |  |
| --- | --- | --- |
| # | Week of​ | ​Lecture Notes |
| ​1 | ​22.09.2025 | ​ |
| ​2 | ​29.09.2025 | ​ |
| ​3 | ​06.10.2025 | ​ Lab 1. MODELING OF DYNAMIC SYSTEM RESPONSE ON UNIT STEP INPUT |
| ​4 | ​13.10.2025 | ​ |
| ​5 | ​20.10.2025 | ​ Lab 2. MODELING OF RLC CIRCUIT RESPONSE ON HARMONIC INPUT |
| ​6 | ​27.10.2025 | 29.10.2025, Holiday |
| ​7 | ​03.11.2025 |  |
| ​8 | ​10.11.2025 | ​Midterms 8-22.11.2025 |
| ​9 | ​17.11.2025 | ​Midterms 8-22.11.2025 |
| ​10 | ​24.11.2025 | ​ |
| ​11 | ​01.12.2025 | Lab 3. Timers |
| ​12 | ​08.12.2025 | ​ |
| ​13 | ​15.12.2025 | ​ |
| ​14 | ​22.12.2025 |  |
| ​15 | ​29.12.2025 | ​​30.12.2023 last day of classes |
| ​16 | ​05.01.2026 | ​Finals 05-20.01.2026 Finals |
| ​17 | ​12.01.2026 | ​Finals |

**PLAGIARISM AND OTHER FORMS OF CHEATING**

Plagiarism is intentionally failing to give credit to sources used in writing regardless of whether they are published or unpublished. Plagiarism (which also includes any kind of cheating in exams) is a disciplinary offence and will be dealt with accordingly. Copying will also be dealt with similarly.

**DEPARTMENTAL POLICY ON TAKING MAKEUP EXAMS**

You can take a make-up for an exam *only if* you are sick and provide a doctor’s report within 3 working days of the exam.

**ANY OTHER USEFUL INFORMATION**

* Lab attendance is mandatory and no make-ups are given for missed labs.
* All students should attend labs and submit the assignment
* NG grade will be given if no grades on the exams/project/labs

19/09/2025