



**Recep Tayyip Erdoğan University**

**Faculty of Engineering and Architecture, Computer Engineering**

**CEN310 - Parallel Programming Course Syllabus**

**Spring Semester, 2024-2025**



## Download

- [PDF](#)
- [DOC](#)
- [SLIDE](#)
- [PPTX](#)



<b>Instructor:</b>	<b>Asst. Prof. Dr. Uğur CORUH</b>
<b>Contact Information:</b>	<a href="mailto:ugur.coruh@erdogan.edu.tr">ugur.coruh@erdogan.edu.tr</a>
<b>Office No:</b>	F-301
<b>Google Classroom Code</b>	Not Used
<b>Microsoft Teams Code</b>	ilpgjzn
<b>Lecture Hours and Days</b>	Friday, 09:00 - 12:00 D-402
<b>Lecture Classroom</b>	D-402 or Online via Google Meet / Microsoft Teams

<b>Instructor:</b>	<b>Asst. Prof. Dr. Uğur CORUH</b>
<b>Office Hours</b>	Meetings will be scheduled via Google Meet or Microsoft Teams using your university account and email. Email requests for meetings are required. To receive a faster response, ensure your email subject begins with <i>[CEN310]</i> , and write clear, concise, formal emails.

<b>Lecture and Communication Language</b>	<b>English</b>
<b>Theory Course Hour Per Week</b>	3 Hours
<b>Credit</b>	4
<b>Prerequisite</b>	None
<b>Corequisite</b>	None
<b>Requirement</b>	Compulsory

## A. Course Description

This course introduces fundamental concepts and practices of parallel programming, focusing on designing and implementing efficient parallel algorithms using modern programming frameworks and architectures. Students will learn to analyze sequential algorithms and transform them into parallel solutions, understanding key concepts such as parallelization strategies, load balancing, synchronization, and performance optimization.

## B. Course Learning Outcomes (Part 1)

After completing this course satisfactorily, a student will be able to:

1. Design and implement parallel algorithms by applying appropriate parallelization strategies and patterns using modern frameworks like OpenMP and MPI
2. Analyze and optimize parallel program performance through proper evaluation of efficiency, scalability, and bottleneck identification
3. Develop parallel solutions using various programming models (shared memory, distributed memory) while effectively managing synchronization and data structures



## B. Course Learning Outcomes (Part 2)

4. Apply parallel computing concepts to solve real-world computational problems using appropriate architectures and tools
5. Evaluate and select appropriate parallel computing approaches based on problem requirements, considering factors such as scalability, efficiency, and hardware constraints

## C. Course Topics

1. Introduction to parallel computing concepts and architecture
2. Parallel algorithm design and performance analysis principles
3. Shared memory programming using OpenMP framework
4. Distributed memory programming with Message Passing Interface (MPI)
5. Performance optimization and profiling tools in parallel systems
6. GPU computing and heterogeneous parallel architecture
7. Advanced parallel programming patterns and synchronization techniques
8. Real-world parallel computing applications and case studies

## D. Textbooks and Required Hardware (Part 1)

This course does not require a specific coursebook. You can use the following books and online resources for reference:

- Peter S. Pacheco, *An Introduction to Parallel Programming*, Morgan Kaufmann
- Michael J. Quinn, *Parallel Programming in C with MPI and OpenMP*, McGraw-Hill
- Barbara Chapman, *Using OpenMP: Portable Shared Memory Parallel Programming*, MIT Press
- Additional resources will be provided during the course

## D. Textbooks and Required Hardware (Part 2)

During this course, you should have:

1. A laptop/desktop with Windows 10 or 11 with the following minimum specifications:
  - Multi-core processor
  - 16GB RAM (recommended)
  - 100GB of free disk space
  - Windows 10 version 2004 and higher (Build 19041 and higher) or Windows 11

## D. Textbooks and Required Hardware (Part 3)

### 2. Required software (all free):

- Visual Studio Community 2022
- Windows Subsystem for Linux (WSL2)
- Ubuntu distribution on WSL
- Git for Windows

## D. Textbooks and Required Hardware (Part 4)

### 3. Development environment setup:

- Visual Studio Community 2022 with:
  - "Desktop development with C++" workload
  - "Linux development with C++" workload
  - WSL development tools
- WSL requirements:
  - Ubuntu on WSL
  - GCC/G++ compiler (installed via apt)
  - OpenMP support
  - MPI implementation (will be installed during class)

## D. Textbooks and Required Hardware (Part 5)

Installation instructions and support for setting up the development environment will be provided during the first week of the course. All programming assignments, classroom exercises, and examinations will be conducted using this setup.

## E. Grading (Part 1)

You will complete one project and two written quizzes throughout the semester. You are expected to submit your Midterm Parallel Implementation Report at the midterm, demonstrating parallel algorithms and performance analysis aligned with your project plan. In the 15th week, you will present and submit your Final Project Implementation Report.

You will take a written quiz in the 7th week and another in the 13th week.



## E. Grading (Part 2)

Assessment	Code	Weight	Scope
Midterm Project Report	MPR1	60%	Midterm
Quiz-1	QUIZ1	40%	Midterm
Final Project Report	MPR2	70%	Final
Quiz-2	QUIZ2	30%	Final

$$GradeMidterm = 0.6MPR1 + 0.4QUIZ1$$

$$GradeFinal = 0.7MPR2 + 0.3QUIZ2$$

$$PassingGrade = (40 * GradeMidterm + 60 * GradeFinal) / 100$$

## E. Grading (Part 3)

Your final passing grade can be improved through the following achievements: (Bonus Points TBD)

$PassingGrade = PassingGrade + Bonus(\text{Tübitak2209A Acceptance, Teknofest Finalist, Hackathon and Similar Finalists})$

## F. Instructional Strategies and Methods

The basic teaching method of this course will be planned to be face-to-face in the classroom, and support resources, homework, and announcements will be shared over Microsoft teams and Github. Students are expected to be in the university. This responsibility is very important to complete this course with success. If pandemic situation changes and distance education is required during this course, this course will be done using synchronous and asynchronous distance education methods. In this scenario, students are expected to be on the online platform, zoom, Microsoft teams or google meets, or meet at the time specified in the course schedule. Attendance will be taken.

## G. Late Homework

Throughout the semester, assignments and reports must be submitted as specified by the announced deadline. Overdue assignments will not be accepted.

Unexpected situations must be reported to the instructor for late homework by students.

## H. Course Platform and Communication

Microsoft Teams Classroom and Github will be used as a course learning management system. All electronic resources and announcements about the course will be shared on this platform. It is very important to check the course page daily, access the necessary resources and announcements, and communicate with the instructor to complete the course with success.

# I. Academic Integrity, Plagiarism, and Cheating

## A. Overview

Academic integrity is one of the most important principles at RTEÜ University. Anyone who violates academic honesty will face serious consequences.

## B. Collaboration and Boundaries

Collaborating with classmates or others to "study together" is a normal aspect of learning. Students may seek help from others (whether paid or unpaid) to better understand a challenging topic or course. However, it is essential to recognize when such collaboration crosses the line into academic dishonesty—determining when it becomes plagiarism or cheating.

## C. Exam and Assignment Guidelines

### 1. Exam Conduct

- Using another student's paper or any unauthorized source during an exam is considered cheating and will be punished.

### 2. Guidelines for Assignments

Many students initially lack a clear understanding of acceptable practices in completing assignments, especially concerning copying. The following guidelines for Faculty of Engineering and Architecture students underscore our commitment to academic honesty. If a situation arises that is not covered below, please consult with the course instructor or assistant.

## **a. What Is Acceptable When Preparing an Assignment?**

### **I. Peer Collaboration and Discussion**

- Communicate with classmates to better understand the assignment.
- Ask for guidance to improve the assignment's English content.
- Share small portions of your assignment in class for discussion.
- Discuss solutions using diagrams or summarized statements rather than exchanging exact text or code.



## a. What Is Acceptable When Preparing an Assignment?

### II. External Resources and Assistance

- Include ideas, quotes, paragraphs, or small code snippets from online sources or other references, provided that:
  - They do not constitute the entire solution.
  - All sources are properly cited.
- Use external sources for technical instructions, references, or troubleshooting (but not for direct answers).
- Work with (or even compensate) a tutor for help, as long as the tutor does not complete the assignment for you.

## **b. What Is Not Acceptable?**

- Requesting or viewing a classmate's solution to a problem before you have submitted your own work.
- Failing to cite the source of any text or code taken from outside the course.
- Giving or showing your solution to a classmate who is struggling to solve the problem.

## J. Expectations

You are expected to attend classes on time and complete weekly course requirements (readings and assignments) throughout the semester. The primary communication channel between the instructor and students will be email. Please send your questions to the instructor's university-provided email address. ***Be sure to include the course name in the subject line and your name in the body of the email.*** The instructor will also contact you via email when necessary, so it is crucial to check your email regularly for communication.

## K. Course Content and Schedule Updates

The course content and schedule may be updated as needed. Any changes will be communicated to students by the instructor.

C. Weekly Lesson Plan (Part 2/4)

C. Weekly Lesson Plan (Part 3/4)

C. Weekly Lesson Plan (Part 4/4)

### Important Time Notes:

- Regular classes: 3-hour sessions on Fridays (09:00-12:00)
- Quiz sessions: Regular 3-hour class period
- Project Review sessions (Week 8 & 15): Full day (09:00-17:00)
- Midterm and Final periods: As scheduled by the university

*End – Of – Syllabus*