CENG577 - Parallel Computing - Fall 2024 v1.0

Lectures: Tuesday 13:40-16:30 in BMB2

Instructor: Murat Manguoğlu

Office:	A407
Phone:	210 5551
Email:	manguoglu@ceng.metu.edu.tr
Web Page:	http://www.ceng.metu.edu.tr/~manguoglu
Office Hours:	Wednesday 14:00-15:00

Motivation:

While the fastest computers are large parallel clusters today, programming them remains a challenge. The best algorithm and their implementation could be quite different than the sequential counterpart on a parallel platform. Programmer needs to consider multiple issues like how to load balance and how to effectively use multiple levels of caches and the communication network. Today parallel computing is everywhere, it is not only the fastest computers, but also from your laptop to cellphone, most devices contain multiple processors. *This course will focus on the architecture of the modern parallel computing platforms and the design, analysis and implementation of parallel algorithms for solving large scale science and engineering problems.*

Outline:

- Introduction and a review of the architectural features of parallel processors
- Communication operations, memory hierarchy and programming models
- Communication networks and their impact on performance
- Task decomposition and design of parallel algorithms
- Parallel dense and sparse matrix computations
- Graphs and graph algorithms for parallelism

Prerequisites:

Programming skills in C/C++. The course is open to graduate students from any department with some background in scientific computing and parallel programming provided there are available seats in the class. With the consent of the instructor, undergraduate students can also register for this course.

Final:

Final will be closed book and notes. Only one page (no larger than A4) and **hand written** cheat sheet is allowed. Time and location will be announced later. We will use gradescope for grading and objections.

Project:

The project can be done individually or with a team of at most 3 people. It will consist of 3-phases: pre-proposal, proposal and final report. The project topics should be chosen so that they require approximately 4-homework equivalent amount of work and they should not have been submitted anywhere else before (as a paper or as work on another course). In the pre-proposal phase, a challenging problem should be determined. If there is a team, it should be formed at this point. The pre-proposals will be due on November 3^{rd}. The proposal should give the existing methods for solving the problem and your brief plan on how to solve the problem. The proposals are due on November 17^{th}. The final reports are due on January 8^{th}.

Delayed submissions are accepted with a penalty of $-5 \times d^2$ where d is the number of days in which the solution is submitted late.

Discussions and communication:

We will use odtuclass.

Attendance and Participation:

Attendance and participation to lectures are encouraged. I may use it in your favor when assigning letter grades.

Grading:

Project	80% (including preproposal-15%, proposal-20% and the final report-45%)
Final	20%
Total	100%

Course Policy and academic honesty:

Discussion of ideas or concepts are allowed and encouraged. However, your codes and reports should be your/your team's own work, without any outside assistance (which obviously includes generative AI tools). If you use a source (online or offline) you are expected to cite it. Violation of these general principles will be handled based on the university regulations and may result in disciplinary action.

Makeup policy:

In case of an official medical or family emergency that prevented you from attending the final, or submitting any of the project reports on time, you should contact the instructor as soon as possible and provide documentation.

References:

Introduction to Parallel Computing, by Grama, Gupta, Kumar, and Karypis, Addison Wesley. 2003

Introduction to High Performance Computing for Scientists and Engineers, by Hager and Wellein, Chapman & Hall/CRC Computational Science. 2010

The Sourcebook of Parallel Computing, Dongarra, Foster, Fox, and Gropp, Kaufmann. 2002

Parallel Programming for Multicore and Cluster Systems, Rauber and Runger, Springer Verlag, 2010.

Introduction to Parallel Computing: A Practical Guide with Examples in C , Petersen and Arbenz, Oxford University Press, 2004.