COMP 429 (01) PARALLEL PROGRAMMING

Fall 2023

1. Course Information

Instructor:	Didem Unat, dunat@ku.edu.tr
KU Credits:	3.00
ECTS Credits:	6.00
Prerequisite(s):	Prerequisite: COMP. 132
Class Location & Meeting Times:	ENG B29 - Tuesday, Thursday 16:00-17:10
PS (Yes/No):	Yes
DS (Yes/No):	No
Lab (Yes/No):	No
Language of Instruction:	English
Office Hours:	Tu 13:00-14:20

Teaching Assistant(s):

	E-Mail	**Phone**	**Office - Office Hours**
•Mohammad Kefah Taha Issa	MISSA18@KU.EDU.TR		ENG 230 **-** Tu 5.30-7 pm Th 1-2.30 pm
•Ilyas Turimbetov	ITURIMBETOV18@KU.EDU.TR		ENG 230 **-** Tu 5.30-7 pm Th 1-2.30 pm

2. Course Description

Fundamental concepts of parallelism. Overview of parallel architectures, multicores, heterogeneous systems, shared memory and distributed memory systems. Parallel programming models and languages. Multithreaded, message passing, data driven, task parallel and data parallel programming. Design of parallel programs, decomposition, granularity, locality, communication, load balancing, and asynchrony. Performance modeling of parallel programs, sources of parallel overheads.

3. Course Overview

Parallel computing paradigms: Shared vs. distributed memory parallelism, SIMD, MIMD and SIMT parallelism, Modern parallel architectures: multi-core and many-core CPUs, General purpose GPUs, heterogeneous computing, Principles of parallel algorithm design: Problem decomposition, load balancing, communication & synchronization, data locality, Parallel programming models: MPI, OpenMP, CUDA, Applications of parallel computing in science, engineering and data analytics, Performance Analysis: speedup, efficiency, performance modeling, Supercomputing.

4. Course Learning Outcomes (CLOs):

CLO #	**Upon successful completion of this course, students will be able to**…
1	Gain knowledge in parallel computer architectures
2	Gain experience in parallel programming on distributed memory and shared memory systems
3	Learn parallelism concepts such domain decomposition, synchronization, barriers, locality, and communication
4	Evaluate and reason about performance of parallel programs and code optimizations

5. Assessment Methods

Method	Description	Weight %
Project	Three Programming Assignments	60.00
Quiz	Three in-person Quizzes	30.00
Participation	Participation, Discussion, and Attendance	10.00
	Total:	100.00

6. Instructional Material and Learning Resources

- An Introduction to Parallel Programming (ISBN: 978-0123742605)
  Author: Peter Pacheco
  Publisher: Kaufmann
  Material Type: Textbook
  Material Status: Required

- Introduction to High Performance Computing for Sci (ISBN: 978-1439811924)
  Author: Georg Hager and Gerhard Wellei
  Publisher: CRC Press
  Material Type: Textbook
  Material Status: Recommended

- Introduction to Parallel Computing (ISBN: 978-0201648652)
  Author: Grama et al
  Publisher: Addison-Wesley
  Material Type: Textbook
  Material Status: Recommended

- Active Use of Course Page on Blackboard: https://ku.blackboard.com/

- KOLT Tutoring: No Service Available

7. Course Schedule

Meeting Times	Subject
WEEK 1	Trends in Computer Architecture, Moore's Law, Taxonomy of Parallel Computers, Memory Models, Parallel Efficiency, Speedup, Scalability, Performance Characterization, Basics of Parallel Programming
WEEK 2-3	Shared Memory Programming with Threads, Race Conditions, Synchronization, Mutual exclusion, False Sharing, Barriers, Locks, Data Dependence, Loop Scheduling, Variable Scope
WEEK 4	Data Access Optimizations, Data Locality Optimizations, Loop Tiling, Loop Fusion, Scalar Optimizations
WEEK 5-6	Task vs Data Parallelism, Combining task and data parallelism, Nested Parallelism
WEEK 7-9	Accelerator programming, GPU architecture, SIMT execution model, GPU memory hierarchy, Programming software-managed memory, Thread divergence, Off-load programming, Streams
WEEK 10	Distributed memory programming, message passing, point-to-point and collective communications, non-blocking communication, network topologies, communication cost modeling
WEEK 11	Regular and Irregular parallel algorithms, real-life application examples, N- body simulations, parallel PDE solvers, Molecular Dynamics Simulations, Deep Learning Applications
WEEK 12	1-sided communication, global address space languages, hybrid programming models
WEEK 12-13	Reducing communication overhead, message aggregation, latency hiding techniques, communication avoiding algorithms, and load balancing
WEEK 13	Topology and affinity in multicore environment

WEEK 14	Future of Parallel Computing, Supercomputing, Top500, History of High Performance Computing, Exascale Systems

8. Student Code of Conduct and Academic Grievance Procedure

Student Code of Conduct

Statement on Academic Honesty with Emphasis on Plagiarism

Academic Grievance Procedure

9. Course Policies

Submission due dates and times are strictly enforced, and no late submissions will be accepted.In order to receive a passing grade, you have to receive a non-zero score from each category above (assignments, quizzes, and participation).There is no makeup for in-class quizzes.

10. Other

You must submit your own work in all assignments, projects and exams. Academic dishonesty includes using other people's words or ideas without acknowledgement, cheating on exams, projects, and homework. In case any of the academic dishonesties are disclosed, disciplinary action will follow.
Any violations of the academic integrity code will be penalized, and may result in failure in the course, suspension, or expulsion from the university.