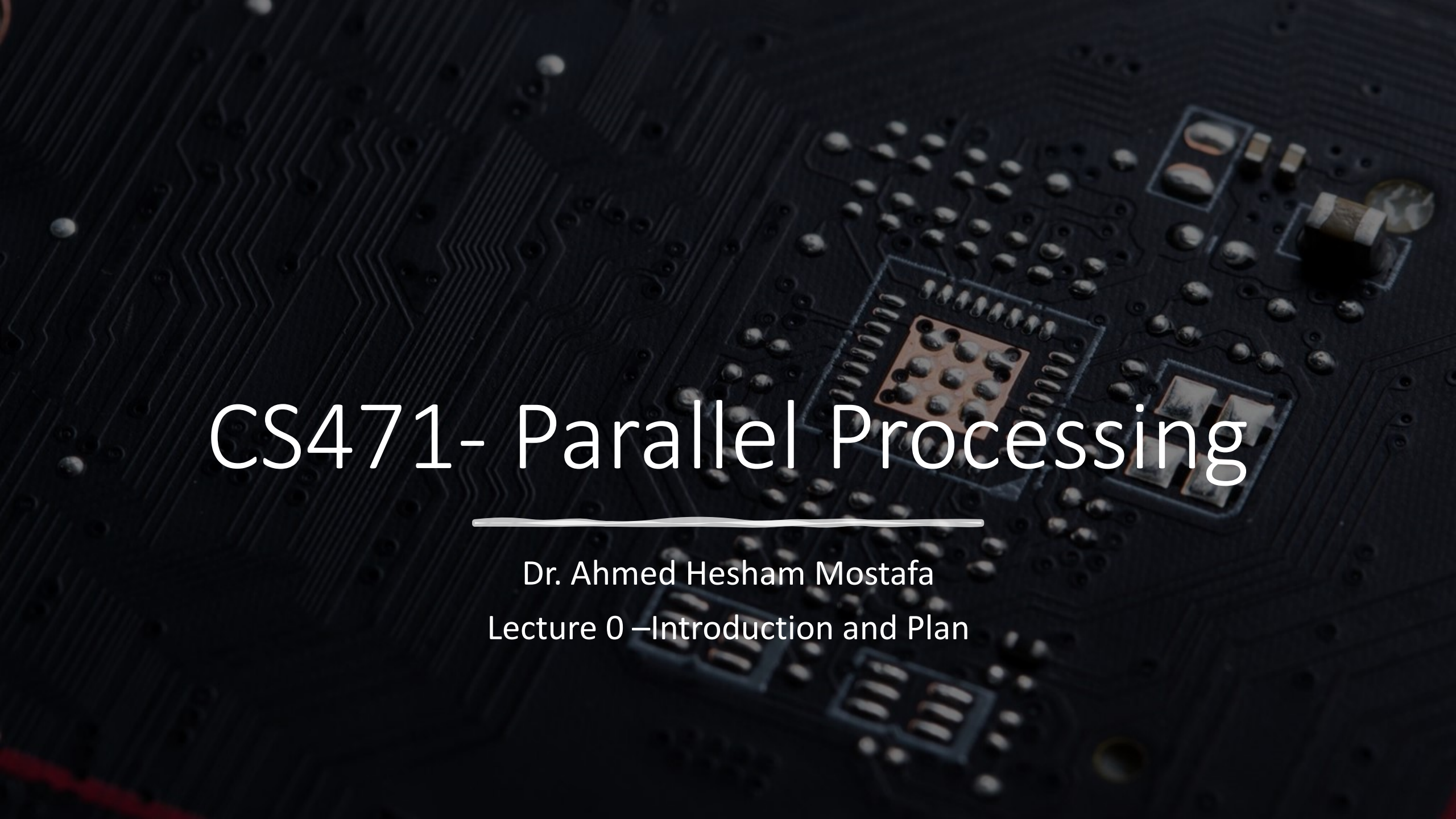

CS471- Parallel Processing

Dr. Ahmed Hesham Mostafa

Lecture 0 –Introduction and Plan

Course Materials & Announcements

- All course material (lecture notes “slides", announcements, assignments, any supplemental notes or documentation), will be made available “posted” online on weekly basis, on Microsoft Teams:
- Team Code :b61bgrv
- https://teams.microsoft.com/l/team/19%3aUdd85fHqtNWc_wNNrEtf5EZsvSbsSbjzfmpx2SIVzyc1%40thread.tacv2/conversations?groupId=630f8f08-b061-4519-b193-83d9a1cbb670&tenantId=aadc0e0a-65ee-471a-99a1-9f86faecbaed

Course Aims and Objectives

- **become very familiar with the core concepts of Parallel Processing**
- **Understand the Instruction-Level Parallelism**
- **Understand arithmetic pipelining**
- **Understand vector processing**
- **Understanding data dependency and data hazard**
- **Be familiar with array processors**
- **Be familiar with multicore architecture and organization**
- **Be familiar with Characteristics of Multiprocessors**
- **Be familiar with Characteristics of Multiprocessors Interconnection Structures**
- **Be familiar with interprocessor Communication and Synchronization**
- **Understand Cache Coherence problem and its condition and solutions**
- **Understand Parallel Algorithms such as sorting and mattix multiplicationtion**

Textbooks

Text#1	Computer System Architecture (3rd Edition) by M. Morris Mano
Text#2	Computer organization and architecture designing for performance (8th edition) by William Stallings
Text#3	Computer Architecture: A Quantitative Approach 5th Edition by John L. Hennessy
Text#4	Advanced Computer Architecture and Parallel Processing by Hesham El-Rewini, Mostafa Abd-El-Barr

Topic covered

Text#1 Chapter 9	Pipeline and Vector Processing
Text#1 Chapter 13	Multiprocessors
Text#2 Chapter 14	Instruction-Level Parallelism
Text#2 Chapter 17	Parallel Processing
Text#2 Chapter 18	Multicore Computers
Text#3 Chapter 3	Instruction-Level Parallelism and Its Exploitation
Text#3 Chapter 4	Data-Level Parallelism in Vector, SIMD, and GPU Architectures
Text#4 Chapter 6	Abstract Models

Grade Policy

- Final 50
- Midterm 30
- Project 20
- Note total of semester work grade is 50
- If the student gets a mark above 50, he will only get 50.

Section

- Section will cover basic Multithreading using Java

Course Project Policy

01

Project groups: Each group will be 4 to 5 students.

02

Project Must be implemented using Java

03

GUI can be implemented using Swing or Javafx

Academic Integrity

You can discuss ideas and methodology for the homework (assignments / sheets) with other students in the course, but you must write your solutions completely independently.

We will be code-checking to assess similar submissions or submissions that use code from other sources.

What is Parallel Processing?

- Parallel processing is a method in computing of running two or more processors (CPUs) to handle separate parts of an overall task. Breaking up different parts of a task among multiple processors will help reduce the amount of time to run a program.
- Any system that has more than one CPU can perform parallel processing, as well as multi-core processors which are commonly found on computers today.
- Multi-core processors are IC chips that contain two or more processors for better performance, reduced power consumption, and more efficient processing of multiple tasks. These multi-core setups are similar to having multiple, separate processors installed in the same computer.
- Parallel processing is commonly used to perform complex tasks and computations. Data scientists will commonly make use of parallel processing for computing and data-intensive tasks.

How parallel processing works

- Typically, a computer scientist will divide a complex task into multiple parts with a software tool and assign each part to a processor, then each processor will solve its part, and the data is reassembled by a software tool to read the solution or execute the task.
- Typically, each processor will operate normally and will perform operations in parallel as instructed, pulling data from the computer's memory. Processors will also rely on software to communicate with each other so they can stay in sync concerning changes in data values. Assuming all the processors remain in sync with one another, at the end of a task, software will fit all the data pieces together.
- Computers without multiple processors can still be used in parallel processing if they are networked together to form a cluster.

References

- SISD,SIMD,MISD,MIMD - A Level Computer Science (learnlearn.uk)
- What is Parallel Processing? (techtarget.com)

Thanks