

CS 5621: Computer Architecture (4)**Catalog Description:**

Advanced concepts in processor and computer system organization and their impact on performance. Exploitation of parallelism, multilevel memory organization, system interconnection, and input-output organization.

Textbook: M. Quinn, *Parallel Programming in C with MPI and OpenMP*, McGraw Hill, 2004
T. Zhang and J. Southmayd, *Teach Yourself C in 24 Hours*, 2nd Ed., Sams, 2000.

Course Goals:

A student who successfully completes this class will be able to:

- Describe the architectures characteristic of modern parallel computing systems.
- Design and implement solutions to large-scale problems using modern parallel computer architectures.
- Understand the challenges posed by the rapid changes in computer architecture taking place today.

Prerequisites by Course & Topic

CS 2521: Computer Organization and Architecture – representation of programs and data, assembly language, addressing.

Major Topics Covered in the Course

- History of Computer Architecture
- Parallel Programming in MPI
- Parallel Design Methodologies
- Cluster Computing
- Multi-Core Computing
- Analysis of Parallel System Performance

Class/Laboratory Schedule: Lecture: 3 hours per week, Laboratory: 1

Course Outcomes

1. Describe the architectures characteristic of modern parallel computing systems.
 - a. Recognize differences between commonly used multiprocessor and multicomputer architectures.
 - b. Understand impact on performance of different memory organizations.
 - c. Identify dominant trends in Top 500 list of supercomputers at any given time.
2. Design and implement solutions to large-scale problems using modern parallel computer architectures.
 - a. Be adept in the use of Foster's Design Methodology for designing solutions to parallel problems.
 - b. Use MPI to implement message passing solutions to be run on very large clusters.
3. Understand the challenges posed by the rapid changes in computer architecture taking place today.
 - a. Appreciate the continued impact of Moore's Law on hardware design.
 - b. Understand the impact of power consumption on individual hardware designs and large data centers.
 - c. Anticipate the great changes that will be ushered in by the rise of multi-core computing.
 - d. Anticipate the great changes that will be ushered in by the use of graphics processors for general purpose computing.

Relationship to Program Outcomes

The prerequisite of CS 5621 is Computer Organization. This course contributes to meeting the following program outcomes:

2. *Students can design, develop, and analyze significant software systems.*
Students design and implement a number of parallel solutions to large scale problems that require the use of hundreds of processors. All course outcomes contribute to this program outcome.
3. *Students understand the fundamentals of computer organization and architecture, data structures and related algorithms, and programming languages.*
Students become skilled at message passing computing, which emphasizes distributed computing and concurrent programming. Students also learn the C programming language. Students learn Flynn's Taxonomy,

and how to classify computer architectures according to it. Students learn how to distinguish between many different kinds of parallel architectures, primarily based on differences in memory organization. Students learn the basics of how state of the art supercomputing systems are designed. All course outcomes contribute to this program outcome.

4. *Students can apply computer science principles and practices to a variety of problems.*

Students design and implement solutions to numeric and I/O intensive problems. All course outcomes contribute to this program outcome.

Assessment Plan for Course:

This course is assessed every third year by the instructor and a course assessment document covering all of the course outcomes and their effect on the program outcomes is prepared.

Estimate CSAB Category Content

	CORE	ADVANCED		CORE	ADVANCED
Data Structures			Computer Organization and Architecture		2
Algorithms		1	Concept of Programming Languages		1
Software Design					

Coordinator/Prepared by: T. Pedersen