

Introduction to Computer Systems

Course Catalog Description:

14:332:434 Introduction to Computer Systems (3)

Multi-core architecture, multi-core CPU management, multi-core job scheduling, thread synchronization, cache coherency, multi-core memory management, multicore file system, I/O

Pre-Requisite Courses:

14:332:331

14:332:351

Co-Requisite Courses:

None

Pre-Requisite by Topic:

C/C++/Java

Data structure

Searching, Sorting

Processor design

Instruction Set Architecture

Memory hierarchy

Textbook & Materials:

Operating System Concepts (9th edition) by Silberschatz, Galvin, Gagne

Parallel Computer Organization and design by Dubois, Annavaram, Stenstrom

Overall Educational Objective:

The course covers the fundamental principles of modern multi-core computer systems. In particular, it introduces the four main components of multi-core computer systems in managing the processor, memory, secondary/tertiary storage and other I/O devices.

Course Learning Outcomes:

A student who successfully fulfills the course requirements will have demonstrated:

1. An ability to understand the multi-core computer architecture
2. An ability to understand what is a process and a thread, and how they are scheduled to execute on multi-core processors
3. An ability to understand how synchronization between processes/threads is implemented
4. An ability to understand how a computer system's performance is assessed and how the performance speed up is achieved in a multi-core system
5. An ability to understand how the memory hierarchy is managed in a multi-core computer system

6. An ability to understand the cache coherency model in a multi-core computer system
7. An ability to understand how a computer's file system works, and whether popular file systems are suitable for multi-core systems
8. An ability to understand how I/O devices are managed
9. An ability to write concurrent programs that involve multiple processes/threads and parallelism
10. An ability to write system-level programs such as file system

How Course Outcomes are Assessed:

- Homeworks (10%)
- Projects (40%)
- Exams (two, 25% each)

Topics Covered in the Course:

1. Modern multi-core computer architecture
2. Process state, context switch, threads on multi-core machines
3. Multi-core CPU scheduling
4. Concurrency, synchronization, communication
5. Virtual memory, paging
6. Cache coherency for multi-core machines
7. File structure, directories
8. File system considerations for multi-core
9. I/O hardware and software
10. I/O challenges for multi-core

N = none S = Supportive H = highly related

Outcome	Level	Proficiency assessed by
(a) an ability to apply knowledge of Mathematics, science, and engineering	H	HW Problems, Exams
(b) an ability to design and conduct experiments and interpret data	N	
(c) an ability to design a system, component or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability	N	
(d) an ability to function as part of a multi-disciplinary team	N	
(e) an ability to identify, formulate, and solve ECE problems	H	HW Problems, Exams
(f) an understanding of professional and ethical responsibility	N	
(g) an ability to communicate in written and oral form	N	
(h) the broad education necessary to understand the impact of electrical and computer engineering solutions in a global, economic, environmental, and societal context	N	
(i) a recognition of the need for, and an ability to engage in life-long learning	S	Home-work
(j) a knowledge of contemporary issues	N	
(k) an ability to use the techniques, skills, and modern engineering tools necessary for electrical and computer engineering practice	H	HW Problems, Exams
Basic disciplines in Electrical Engineering	H	HW Problems, Exams
Depth in Electrical Engineering	S	HW Problems, Exams
Basic disciplines in Computer Engineering	H	
Depth in Computer Engineering	H	
Laboratory equipment and software tools	S	HW Problems, Mid-Term
Exams		
Variety of instruction formats discussions	S	Lecture, office hour