CSE 6333 (Approved): Distributed Algorithms

Course Description

Fundamental concepts in distributed computing; algorithms for distributed control and data; impossibility and limits; algorithms for fault-tolerance; specification, design and verification of distributed programs.

Prior Course Number: CSE 763
Transcript Abbreviation: Distributed Algs
Grading Plan: Letter Grade
Course Deliveries: Classroom
Course Levels: Graduate
Student Ranks: Senior, Masters, Doctoral
Course Offerings: Spring
Flex Scheduled Course: Never
Course Frequency: Every Year
Course Length: 14 Week
Credits: 3.0
Repeatable: No
Time Distribution: 3.0 hr Lec
Expected out-of-class hours per week: 6.0
Graded Component: Lecture
Credit by Examination: No
Admission Condition: No
Off Campus: Never
Campus Locations: Columbus
Prerequisites and Co-requisites: CSE 6431 or CSE 760
Exclusions: Not open to students with credit for CSE 763
Cross-Listings:

The course is required for this unit's degrees, majors, and/or minors: No
The course is a GEC: No
The course is an elective (for this or other units) or is a service course for other units: Yes

Subject/CIP Code: 14.0901
Subsidy Level: Doctoral Course

Programs

Abbreviation	Description
BS CSE	BS Computer Science and Engineering
MS CSE	MS Computer Science and Engineering
PhD CSE	PhD Computer Science and Engineering

Course Goals

Master the principles behind several specific classes of distributed algorithms for solving particular problems.
Master reading and understanding distributed algorithms.
Be familiar with writing and designing distributed programs.
Be familiar with the analysis and verification of distributed programs.

Course Topics

Topic	Lec	Rec	Lab	Cli	IS	Sem	FE	Wor
Global Time and Order: logical clock, vector clock, causal order and broadcast, clock synchronization algorithms.	5.0							
Global State and Predicate Detection: consistency, snapshots, termination detection algorithms.	4.0							
Programming Notation and Logic: syntax and semantics for distributed programs, safety and progress properties, a UNITY-style temporal logic, proofs of program properties, examples, fairness.	9.0							
Fault-Tolerance: consensus, impossibility of robust consensus, leader election, atomic commitment, Byzantine agreement, Paxos, self-stabilization, distributed reset.	9.0							
Wait-freedom: atomic register algorithms.	2.0							
Distributed and Concurrent Data Structures: queues, skip lists, hash sets, linearizability.	5.0							
Scalability: local, geometric, dense network, small world algorithms.	4.0							

Grades

Aspect	Percent
Assignments	35%
Midterm	30%
Final	35%

Representative Textbooks and Other Course Materials

Title	Author
Elements of Distributed Computing	Vijay K. Garg

ABET-EAC Criterion 3 Outcomes

Course Contribution		College Outcome
**	a	An ability to apply knowledge of mathematics, science, and engineering.
	b	An ability to design and conduct experiments, as well as to analyze and interpret data.
***	c	An ability to design a system, component, or process to meet desired needs.
	d	An ability to function on multi-disciplinary teams.
**	e	An ability to identify, formulate, and solve engineering problems.
	f	An understanding of professional and ethical responsibility.
	g	An ability to communicate effectively.
*	h	The broad education necessary to understand the impact of engineering solutions in a global and societal context.
	i	A recognition of the need for, and an ability to engage in life-long learning.
*	j	A knowledge of contemporary issues.
***	k	An ability to use the techniques, skills, and modern engineering tools necessary for engineering practice.

BS CSE Program Outcomes

Course Contribution		Program Outcome
**	a	an ability to apply knowledge of computing, mathematics including discrete mathematics as well as probability and statistics, science, and engineering;
	b	an ability to design and conduct experiments, as well as to analyze and interpret data;
***	c	an ability to design, implement, and evaluate a software or a software/hardware system, component, or process to meet desired needs within realistic constraints such as memory, runtime efficiency, as well as appropriate constraints related to economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability considerations;
	d	an ability to function on multi-disciplinary teams;
**	e	an ability to identify, formulate, and solve engineering problems;
	f	an understanding of professional, ethical, legal, security and social issues and responsibilities;
	g	an ability to communicate effectively with a range of audiences;
*	h	an ability to analyze the local and global impact of computing on individuals, organizations, and society;
	i	a recognition of the need for, and an ability to engage in life-long learning and continuing professional development;
*	j	a knowledge of contemporary issues;
***	k	an ability to use the techniques, skills, and modern engineering tools necessary for practice as a CSE professional;
**	l	an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution;
**	m	an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices;
*	n	an ability to apply design and development principles in the construction of software systems of varying complexity.

Prepared by: Anish Arora