Basic Course Information CS 5350

Course Title: Parallel and Distributed Algorithms

Units: 3

C/S Classification #: C-2

Component (select one): Lecture

Instructional Mode (select all appropriate choices): Face-to-Face and web-assisted

Grading Basis (select one): Graded only

Repeat Basis (select one): May be taken only once

Cross listed Course (if offered with another department):

Dual-listed Course (if offered as lower/upper division or undergraduate/graduate):

Major course/Service course/GE course (select all appropriate choices): Major course

Prepared by: Gilbert Young

I Catalog Description

Models of parallel and distributed computation. Design and analysis of algorithms for parallel and distributed systems. Basic techniques, classic problems. Parallel and distributed complexity classes. Hardware and software issues involved in parallel and distributed problem solving.

II Required Coursework and Background

Pre-requisite(s): CS 3310 or consent of instructor

III Expected Outcomes

On successful completion of this course, students will be able to:

1. Gain an appreciation for the potential and limitation of parallel processing

1. Learn parallel models and their realization in hardware
2. Comprehend techniques for exploiting parallelism that serves as building blocks for algorithm design paradigms
3. Analyze the computational complexity of a parallel algorithm

Outcomes of this course will build student capacity in each of the following areas as defined by programmatic objectives for the computer science major.

P-SLO 1. An ability to frame and model real-world problems that can be addressed computationally, and evaluate multiple computational approaches and select the most appropriate one.

IV Instructional Materials

Texts may vary with instructor and over time. Examples of possible texts include:

C. Xavier and S. S. Iyengar, Introduction to Parallel Algorithms, Wiley, 1998

Joseph JaJa, An Introduction to Parallel Algorithms, Addison-Wesley, 1992

Selim G. Akl, The Design and Analysis of Parallel Algorithms, Prentice-Hall, 1989

F.T. Leighton, Introduction to Parallel Algorithms and Architectures: Arrays, Trees, Hypercubes, Morgan Kaufmann Publishers, 1991

V Minimum Student Material

Course textbooks

VI Minimum College Facilities

Computer laboratories, Library, Blackboard, classroom with a projection system

VII Course Outline

The need for parallel computers

Parallelism, the PRAM and other models

Analyzing parallel algorithms

Some PRAM algorithms

Real parallel machines – MIMD, SIMD, MISD

Parallel architectures and operating systems

Language features to support parallelism

Parallel issues: Processor scheduling, shared memory, message passing

Parallel merging and sorting

Parallel computational complexity

VIII Instructional Methods

Lecture

Problem-solving

Discussion

Project-based learning

IX Evaluation of Outcomes

A. Student Assessment

i homework assignments

ii exam

iii term paper

B. Meaningful Writing Assignment

Students shall produce written solutions or proofs to problems that are assigned as homework and explain their reasoning. Students are required to write a term paper after conducting a research study on an advanced topic of parallel and distributed algorithms.

C. A Matrix of Course Student Learning Outcomes vs Methods of Assessment

If the course is being evaluated for accreditation purposes, approved department accreditation assessment tools will additionally be utilized.

|  |  |  |  |
| --- | --- | --- | --- |
| Course Learning Outcomes | Methods of Assessment | | |
| Homework Assignments | Team Paper | Exams |
| Gain an appreciation for the potential and limitation of parallel processing | x | x |  |
| Learn parallel models and their realization in hardware | x | x |  |
| Comprehend techniques for exploiting parallelism that serves as building blocks for algorithm design paradigms | x | x | x |
| Analyze the computational complexity of a parallel algorithm | x | x | x |