COURSE DESCRIPTION

Dept., Number	CSci 490	Course Title	Special Topics: Parallel Programming, Summer 2006
Semester hours	3	Course Coordinators	H. Conrad Cunningham, Professor Jason G. Hale, Adjunct Instructor, staff member of Mississippi Center for Supercomputing Research (MCSR)

Current Catalog Description

Study of topics in computer science according to the interests of the instructor and students.

Textbook

Michael Quinn. *Parallel Programming in C with MPI and OpenMP*, 1st Edition, McGraw-Hill, 2004.

References

Mississippi Center for Supercomputing Research (MCSR) website: www.mcsr.olemiss.edu

Course Outcomes

Upon successful completion of this course, the students are able to:

1. develop parallel programs using collective and point-to-point message-passing techniques;
2. develop shared-memory parallel programs using threads;
3. assess the performance of parallel programs in terms of speed-up and efficiency;
4. determine when to use message-passing and when to use shared-memory programming techniques;
5. decompose and balance workload between parallel processes to maximize performance.

Relationship between Course Outcomes and Program Outcomes

This is an elective course taken by undergraduate computer science students to enrich their programs. Course outcomes 1 through 5 all contribute to program outcomes (c) and (k).

Prerequisites by Topic

1. Basic data structures and algorithms (CSci 112, 211)
2. Fundamental computer architecture concepts (CSci 223)

Major Topics Covered in the Course

1. Review of Unix concepts and commands (3 hours)
2. Review of the C language (3 hours)
3. MPI collecting message-passing techniques (3 hours)
4. MPI synchronous point-to-point message passing (3 hours)
5. MPI asynchronous point-to-point message passing (3 hours)
6. Methods of task and data decomposition (3 hours)
7. Analyzing potential and actual performance of parallel programs (3 hours)
8. Static load balancing (2 hours)
9. Dynamic load balancing (centralized, decentralized, work farm, termination) (4 hours)
10. Shared-memory programming with standard Unix processes (3 hours)
11. Shared-memory programming with PThreads (3 hours)
12. Shared-memory programming with MPI (3 hours)
13. Exams (3 hours)

Assessment Plan for the Course

This is an elective course offered on this topic three times in the past six years. An offering typically has 2 mid-term examinations and 5 homework assignments. Outcome 1 is assessed in homework assignments 2 and 3 and exam 1; outcome 2 is assessed in homework assignment 5 and exam 2; outcome 3 is assessed in homework assignment 1 and exam 1; outcome 4 is assessed in homework assignment 4 and exam 2. The course coordinator and instructor evaluate the student performance informally and make changes to the course content, organization, and pedagogy as appropriate for future offerings of the topic.

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)

Estimate Curriculum Category Content (Semester hours)

Area	Core	Advanced	Area	Core	Advanced
Algorithms		1.5	Software design		1.5
Data structures			Concepts of programming languages		