CSCI 3366 (Introduction to Parallel and Distributed Processing), Fall 2005

Syllabus

1 Course description

This course is an introduction to parallel and distributed processing, including both the theory and the application of parallel-processing concepts. Course content will include discussions of different types of parallel machines and machine models, the design and analysis of parallel algorithms, and the development of parallel programs.

The objectives of this course include, but are not limited to, the following:

- Learning fundamental parallel processing concepts.
- Learning parallel algorithm design.
- Learning the basics of parallel machine structure.
- Programming using message-passing (e.g., using MPI).
- Programming using threads (e.g., using the POSIX threads library, OpenMP, and/or Java).

2 Basic information

Class meeting times and location

- MWF 11:30am – 12:20pm, HAS 340

Prerequisites

- CSCI 2320, or consent of instructor

Instructor and contact information

- Dr. Berna Massingill
- Office: HAS 201L
- Office phone: (210) 999-8138
- Web page: `http://www.cs.trinity.edu/~bmassing/`
- E-mail: `bmassing@cs.trinity.edu`
- Office hours:
  - Monday/Wednesday 3:30pm – 5:30pm in HAS 340
  - Tuesday/Thursday 2pm – 5pm in HAS 201L

  In addition to these scheduled office hours, you are welcome to drop by and see if I am in my office and free to talk, or you can make an appointment by calling me or sending me e-mail. If I am not in my office during scheduled office hours, I should be somewhere in the building (perhaps in one of the labs helping another student), and there will usually be a note on my door saying where to find me.
- Open lab hours: Notice that some of my office hours will be in HAS 340. This is "open lab", intended as a time when you can come work on your homework with someone available to help with any questions or problems.

3 Course materials

Textbook

- Timothy G. Mattson, Beverly A. Sanders, and Berna L. Massingill. *Patterns for Parallel Programming.* Addison-Wesley, 2004.

Web page

Most course-related information (this syllabus, homework and reading assignments, etc.) will be made available via the Web. The course Web page is a starting point for Web-accessible course material; you can find it linked from my home page (`http://www.cs.trinity.edu/~bmassing`) or directly at `http://www.cs.trinity.edu/~bmassing/Classes/CS3366_2005fall/`.

Other references

- Gregory R. Andrews. *Multithreaded, Parallel, and Distributed Programming.* Addison-Wesley, 2000. A more theory-oriented treatment.
- M. Ben-Ari. *Principles of Concurrent and Distributed Programming.* Prentice Hall, 1990. A more theory-oriented treatment.
- Rohit Chandra, Leonardo Dagum, Dave Kohr, Dror Maydan, Jeff McDonald, and Ramesh Menon. *Parallel Programming in OpenMP.* Morgan Kaufmann, 2000.
- K. Mani Chandy and Jayadev Misra. *Parallel Program Design: A Foundation.* Addison Wesley, 1989. A much more theory-oriented treatment, co-authored by my thesis advisor.
- Jack Dongarra, Ian Foster, Geoffrey Fox, Ken Kennedy, Andy White, Linda Torczon, and William Gropp, editors. *The Sourcebook of Parallel Computing.* Morgan Kaufmann, 2002.
- Ian Foster. *Designing and Building Parallel Programs.* Addison Wesley, 1995.
- William Gropp, Ewing Lusk, and Anthony Skjellum. *Using MPI: Portable Parallel Programming with the Message-Passing Interface.* The MIT Press, 1999. Fifth Printing.
- William Gropp and Marc Snir. *MPI: The Complete Reference.* MIT Press, 2nd edition, 1998.
- Peter Pacheco. *Parallel Programming with MPI.* Morgan Kaufmann, 1996.
- Michael J. Quinn. *Parallel Programming in C with MPI and OpenMP.* McGraw-Hill, 2004. The textbook I would use if mine didn't exist.
- Barry Wilkinson and Michael Allen. *Parallel Programming: Techniques and Applications Using Networked Workstations and Parallel Computers.* Prentice Hall, 1999. A book Dr. Eggen and I have used in the past as a textbook for this course.

4 Course requirements

Grading

Grades in this course will be determined by the results of several homework assignments, a project, and class attendance/participation, weighted as follows.

Component	**Maximum points**
Homework	about 200
Project	100
Class participation	20

Numeric grades will be calculated as a simple percentage, by dividing total points earned on the above components by total points possible. These numeric grades will then be converted to letter grades based on a curve, but in no case will the resulting letter grades be worse than students would receive based on the following scheme.

Numeric grade	**Letter grade**
90 – 100	A
80 – 89	B
70 – 79	C
60 – 69	D
0 – 59	F

Homework assignments

Several homework assignments will be required for successful completion of this course; most if not all will involve programming. Detailed requirements, including due dates and times, will be provided as part of each assignment. Programming problems will be coded using suitable parallel languages or libraries (e.g., MPI or Java) as specified in individual assignments.

Project

As part of the course, students must also complete a significant project approved by the instructor and present it to the class; students may work individually or in groups of two. Detailed requirements for the project will be described separately and will include program code, a short written report, and a presentation to the class.

Notice that although there are no exams in this course, we will use the time scheduled for a final (December 13 at 8:30am) for project presentations. Please plan accordingly.

Attendance

Regular class attendance is strongly encouraged; class participation grades will be based largely on attendance.

E-mail

Course-related announcements will sometimes be made by sending e-mail to the Trinity e-mail addresses of all registered students. Students are strongly encouraged to read mail sent to their Trinity addresses frequently. An archive of such announcements will be provided via the course Web page.

Late and missed work

Unless otherwise stated for a particular assignment, homework will be accepted up to one class period late, *but no more*, at a penalty of 10 percent off per working day. This penalty may be waived or additional time allowed *at the instructor's discretion* in cases of illness or conflict with a university-sponsored activity or religious holiday.

If you have unusual circumstances (as we all sometimes do), please discuss these with the instructor as far in advance as possible.

Collaboration and academic integrity

Unless otherwise specified, all work submitted for a grade (homework assignments and projects) must represent the student's own individual effort. For students covered by the Academic Honor Code, unless otherwise stated all submitted work (homework and projects) will be considered pledged work. Discussion of homework assignments and course material among students is encouraged, but not to the point where detailed answers are being written collectively. Graded papers and sample solutions from previous years (for this course, homeworks) are off limits. Answers that are identical beyond coincidence (either to another student's work or to a sample solution from a previous year) will be considered to be in violation of Trinity's Academic Integrity Policy or Academic Honor Code, whichever applies, and *will result in disciplinary action.* You are responsible for the security of your work, both electronic and hard copy.