# CSCI 3366 (Introduction to Parallel and Distributed Processing), Spring 2001

## Review for Exam 1

*Note:* The HTML version of this document may contain hyperlinks. In this version, hyperlinks are represented by showing both the link text, formatted like this, and the full URL as a footnote.

## 1 Format of the exam

The exam will be in class February 22. You will have 75 minutes. You may use your textbook and any notes or papers you care to bring, but you may not use other books, a calculator or computer, or each other's papers.

The following are some kinds of questions that might be on the exam. It is *not necessarily an exhaustive list* of all types of questions on the exam, but should give you an idea of what to expect.

- Multiple-choice and/or short-answer questions.
- Questions in which you are given a pseudocode algorithm or some MPI-based code and asked what it does, whether it works, how to fix it, etc.
- Questions in which you are asked to write a pseudocode algorithm or some MPI-based code to accomplish a given task. You may be given some of the code and asked to "fill in the blanks", or you may be given descriptions of functions to use in your solution.

## 2 Material to review

### 2.1 Reading

You are also responsible for material in the assigned reading (chapters 1 through 4 of the textbook), except as noted below.

- In chapter 1, you may skip sections 1.3.2 and 1.3.3.
- In chapter 2, you may skip sections 2.2.2, 2.3.4, and 2.4.2.
- In chapter 3, you may skip section 3.2.3.
- In chapter 4, you may skip section 4.2.2.

You also are not responsible for understanding in detail the mathematics behind the computations (e.g., you do not need to know, for the program of Homework 2, exactly what the numbers being computed mean, only how to compute them in parallel).

## 2.2 Topics

You are also responsible for all material covered in class. Below is a list of topics we have covered; it is *not necessarily exhaustive*, but should give you an idea of what topics I consider most significant.

- Why parallel programming?
- Shared-memory and distributed-memory (message-passing) paradigms.
- "Speedup factor" and Amdahl's law.
- Basics of message-passing programming (processes, point-to-point communication, how to deal with lack of shared memory).
- Estimating execution time.
- "Embarrassingly parallel" computations — key idea (independent tasks), examples.
- Divide-and-conquer computations — key idea, examples.