CSCI-UA.0480-051: Parallel Computing
Practice Exam (October 26th, 2023)
Total: 100 points

Important Notes- READ BEFORE SOLVING THE EXAM

• If you perceive any ambiguity in any of the questions, state your assumptions clearly and solve the problem based on your assumptions. We will grade both your solutions and your assumptions.
• This exam is take-home.
• You have up to 24 hours to complete this exam.
• Your answers must be very focused. You may be penalized for giving wrong answers and for putting irrelevant information in your answers.
• Your answer sheet must be organized neatly.

Honor code (copy and paste to the first page of your exam)

"I understand the ground rules and agree to abide by them. I will not share answers or assist another student during this exam, nor will I seek assistance from another student or attempt to view their answers."

Problem 1

a. [10] Briefly explain the difference between shared memory and distributed memory parallel programming models.

_____
_____

b. [10] What are the advantages and disadvantages of using threads versus processes for parallel programming?

_____
_____

Problem 2

[20] A program needs to process an array of 1000 integers. Describe how you would parallelize this task using OpenMP, including code snippets illustrating the key concepts. Assume each integer requires an independent computation.

_____
_____
_____
_____

Problem 3

[20] Explain Amdahl's Law and its implications for the scalability of parallel programs. Provide a specific example to illustrate its effect.

_____
_____
_____
_____

Problem 4

a. [15] You are given the following task graph:

A -> B, C

B -> D

C -> D

D -> E

Each task takes 1 unit of time. Draw a possible execution schedule on two processors, minimizing the overall completion time. Indicate which processor executes which task and at what time.

_____

_____

_____

b. [5] What is the critical path in the task graph from part (a)?

_____

Problem 5

[20] Describe a common deadlock scenario in parallel programming. Explain how you could prevent this scenario using appropriate synchronization mechanisms. Provide code examples (pseudocode is acceptable) to illustrate both the deadlock and its prevention.

_____

_____

_____

_____

_____