CX 4220/CSE 6220 High Performance Computing
Spring 2023
Homework 1
Due Wednesday, January 25

Adhere to the following guidelines while working on and submitting the homework

- You are strongly encouraged to be concise in answering homework problems, and type up your solutions (preferably using LaTeX).
- It is your responsibility to ensure that your solutions are legible. You will risk losing points if your solutions are illegible to the TAs.
- Submissions are due 11:59PM EST. The deadline for distance learning students is one week after the date on the homework. Late homeworks are not accepted.
- Your submission MUST be made in PDF format. Specify your name and GT username at the top. Do not put your GTID.

1. (5 points) Let S be an array containing unordered elements and you are to develop an algorithm to find the location of a given $x \in S$. Measure the run-time in terms of the number of comparisons required. Average case analysis should be done rigorously by computing the average over all possible inputs, taking each input to be equally likely.

   (a) Compute the best case, worst case, and average case run-times of the serial algorithm.

   (b) To solve the problem in parallel, S is evenly distributed on p processors. For simplicity, only count the number of comparisons required as a measure of the run-time, and assume p divides n. Compute the best case, worst case, and average case run-times of the parallel algorithm.

   (c) Compute the best case speedup, worst case speedup, and average case speedup. State your observations.

2. (5 points) The following information is given on a parallel algorithm:

$$\begin{aligned} T(n,1) &= n^2 \\ T(n,p) &= \frac{n^2}{p} + n, \qquad p \leq n^2 \end{aligned}$$

   Write the expression for speedup. What happens to the speedup if n is fixed and p is continually increased? What happens to the speedup if the problem size n is kept proportional to p, the number of processors (i.e., $n = kp$ for some constant k)? Comment on your answers.

3. (5 points) We are given two parallel algorithms for solving a problem of size n.

$$T(n, n^2) = \sqrt{n}$$

$$T(n, n) = n$$

Which algorithm will run faster on a machine with p processors? Do not make any particular assumption on the value of p. Instead, consider all possible values of p.

4. (5 points) The following information is given on a parallel algorithm:

$$\begin{array}{lcl} \text{Serial execution time} & = & n^2 \\ \text{Parallel execution time} & = & \frac{n^2}{p} + pn \\ \text{Memory required per processor} & = & \frac{n}{p} \end{array}$$

   (a) Find the largest number of processors (as a function of n) that can be used while being optimally efficient. (3 points)

   (b) Assuming that memory available per processor is fixed, can we scale the number of processors as a function of n according to the equation derived above? (2 points)

5. (5 points) The complexity of the best sequential algorithm for solving a problem is $\Theta(n^2)$. A parallel algorithm designed for solving the same problem has a complexity given by

$$T(n, p) = \Theta\left(\frac{n^2}{p} + \frac{n}{\sqrt{p}} \log p\right) \qquad p \leq n^2$$

   Find the maximum number of processors as a function of n that can be used such that the algorithm runs with maximum possible efficiency.