CX 4220/CSE 6220 High Performance Computing
Spring 2023
Homework 3
Due Monday, March 6

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) Draw a 8-element bitonic sorting circuit to sort elements in **non-increasing** order. Use horizontal lines for the numbers and vertical lines to denote comparison operations. Label the comparison operations as $\uparrow$ or $\downarrow$ where the direction of the arrowhead indicates the destination of the smaller element. Illustrate how the input $15, 2, 7, 4, 17, 3, 9, 1$ is sorted using the diagram.

2. (5 points) The *Bitonic Split* operation defined in class assumes that the bitonic sequence has an even length. Extend this operation to bitonic sequences of odd length. Now, show how the bitonic sequence $5, 3, 4, 7, 10, 14, 17, 13, 8$ can be converted into a sorted sequence using repeated bitonic split operations.

3. (5 points) Give an algorithm to merge two sorted sequences of lengths m and n, respectively. You may assume that the input is an array of length $m + n$ with one sequence followed by the other, distributed across processors such that each processor has a subarray of size $\frac{m+n}{p}$. What are the computation and communication times for your algorithm? (You may assume the use of any permutation-style communication, not necessarily hypercubic.)

4. (5 points) Two algorithms are designed for the **All-to-all** communication primitive, and have the following runtimes:

$$\begin{aligned} \text{Algorithm 1} &: \quad \Theta\left(\tau \log p + \mu m p \log p\right) \\ \text{Algorithm 2} &: \quad \Theta\left(\tau p + \mu m p\right) \end{aligned}$$

Determine which algorithm runs faster asymptotically as a function of the message size.

5. (5 points) Consider a tree of n nodes and having a bounded degree (i.e., the number of children per node is bounded by a constant). The tree is stored in an array A of size n. Each node also contains the indices at which its parent and children are stored in the array. The array is distributed among processors using block decomposition, i.e., P_i contains $A[i\frac{n}{p}], A[i\frac{n}{p} + 1], \ldots, A[(i+1)\frac{n}{p} - 1]$. For each of the following operations, which communication primitive will you use and what is the runtime?

   (a) Each node in the tree has $O(1)$ sized data that should be sent to its parent.

   (b) Each node in the tree has $O(1)$ sized data that should be sent to all its children.