CSE Ph.D. Qualifying Exam, Spring 2021
High Performance Computing

Instructions:

Please answer three of the following four questions. All questions are graded on a scale of 10. If you answer all four, all answers will be graded and the three lowest scores will be used in computing your total.

Questions:

1. Consider function $f(x)$ which is defined as follows: $f(1) = 1$. For $x > 1$,

$$f(x) = \begin{cases} f(\frac{x}{2}) & \text{if } x \text{ is even} \\ f(3x+1) & \text{if } x \text{ is odd.} \end{cases}$$

   It is conjectured that for all integer values of x, $f(x)$ resolves to 1. Design efficient parallel strategies to test this conjecture $\forall x \leq N$.

   Please provide a brief description of key ideas, and not pseudocode.

2. Consider the multiplication of two matrices A and B, where A has dimensions $m \times k$ and B has dimensions $k \times n$. Suppose that you are given p interconnected distributed memory compute nodes and that communication of 1 floating point word between two nodes is 100 times more expensive (in wall-clock time) than 1 floating point operation.

   (a) Given m, k, n, and p, derive an algorithm that minimizes the total time for performing the multiplication. You do not need to use all p compute nodes. Your answer is expected to be different depending on the values of m, k, n, and p. Be sure to describe how the matrices A and B are initially distributed on the compute nodes, and also how the result is distributed.

   (b) Now assume that each compute node has a local memory of M words that is assumed to be small. Reading and writing from local memory is 10 times more expensive than 1 floating point operation. Prove a lower bound for the computation time in terms of m, k, n, p, and M. Again, you can use up to p compute nodes.

   Hint: The SUMMA and Cannon algorithms are not solutions to the above problems.

3. Suppose we have a hypercube network of $N = 2^{m+n}$ processors. Assume that a matrix $A \in \mathbb{R}^{2^m \times 2^n}$ is distributed on the network, so that $A_{i,j}$ is in the memory of processor p such that $(p)_2 = ((i)_2, (j)_2)$, where $(k)_2$ is the binary representation of the number k and (x, y) is the concatenation of bits. For example if $m = 3$ and $n = 2$, $A_{5,1}$ is assigned to processor 21, because $(21)_2 = 10101 = (101, 01) = ((5)_2, (1)_2)$.

   Write an algorithm to store $B = A^T$, so that $B_{i,j} = A_{j,i}$ and B follows the same memory layout rule as A.

   Your algorithm should assume each process can only store a constant number of matrix element at a time. For full points, you should prove that your algorithms uses the fewest rounds of communication possible.

4. Given a binary matrix b with M rows, N columns, of Boolean values (i.e., 0 and 1). Design and analyze an efficient parallel algorithm that finds the largest (most elements) rectangular subarray containing all ones.

   Be sure to describe how the matrix b is initially distributed on the compute nodes, and where the result is stored.