CSE Qualifying Exam, Fall 2023: High Performance Computing

This is a **closed book** exam. No books or notes are allowed.

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

1. **Folding to align the stars.** Consider a binary string S (0 and 1 characters) of length n. Suppose we wish to *fold* this string onto the plane so as to maximize the number of 1 values aligned in a certain way, as described below and illustrated in Figure 1.

   - The figure shows a string of length $n = 30$ with solid stars representing 1 values and hollow circles representing 0 values. The numbers inside circles are the corresponding string positions.
   - When laid out in the plane, the string may only go straight down (or up), then turn and move one horizontal step, then go straight up (or down), without crossings. In the figure, folds occur at the pair-positions (5, 6), (12, 13), (17, 18), (21, 22), and (26, 27).
   - A pair of 1-values are *aligned* when they are adjacent horizontally *and* nonconsecutive. In the figure, the pairs at positions (3, 8), (10, 15), (15, 20), and (24, 29) are aligned. (If 5 and 6 had been stars, they would *not* be considered aligned.)

   Give an efficient parallel algorithm to determine the folding that maximizes the number of aligned stars (1 values). Take the output to be the sequence of folds as pair-positions.

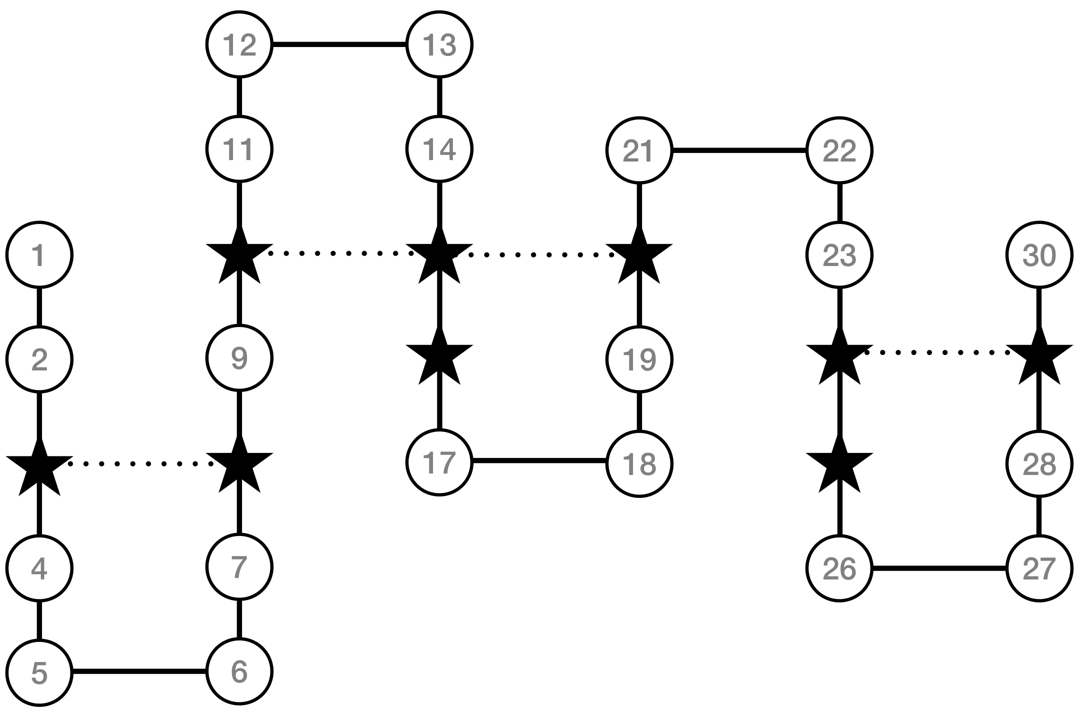

Figure 1: An example of a folded string of length 30 whose “aligned stars” occur at pair-positions (3, 8), (10, 15), (15, 20), and (24, 29).

2. Suppose we wish to compute the solution of an $n \times n$ tridiagonal linear system

$$\begin{bmatrix} d_0 & a_0 & & & & \\ b_0 & d_1 & a_1 & & & \\ & b_1 & d_2 & a_2 & & \\ & & \ddots & \ddots & \ddots & \\ & & & b_{n-2} & d_{n-2} & a_{n-2} \\ & & & & b_{n-1} & d_{n-1} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_{n-2} \\ x_{n-1} \end{bmatrix} = \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ \vdots \\ b_{n-2} \\ b_{n-1} \end{bmatrix},$$

where the a_i's, b_i's, c_i's, and d_i's are given and the x_i's are to be computed. Design a shared-memory parallel algorithm for this task and analyze its work and run-time. For full credit, your algorithm must be work-optimal (i.e., $\mathcal{O}(n)$ operations).

3. Consider the following model for a dynamic load balancing scheme. Let P denote the number of processors. When a processor is available do work, it executes a critical region that assigns the processor v units of work, where v is constant. The critical region requires 11 processor cycles to execute and the assigned work requires $43v$ cycles to execute. Assume that the total number of units of work is N, where N is large, but finite. When a processor enters the critical region and there is no work remaining, the processor halts after leaving the critical region. To be explicit, only one processor can execute the critical region at any time, and a processor must wait if another processor is in the critical region.

   For the questions below, please explain how you came up with your answers.

   (a) (25%) How should v be chosen?

   (b) (25%) Give a mathematical condition that tells us when there is always at least one processor that is waiting to enter the critical region.

   (c) (25%) Give an expression for the speedup that is achieved in different circumstances.

   (d) (25%) Give a recommendation for the number of processors P that should be used for this computation.

4. A new static interconnection network is defined as follows: Let the number of processors $P = 2^d$, and the rank of each processor is denoted by a d bit binary string just as in the hypercube. However, two processors are connected if and only if the bit representations of their ranks differ in exactly k bits.

   (a) (10%) Show that for even k, the network is disconnected.

   (b) (50%) Let k be odd and assume d is sufficiently larger than k. Find a path between two processors whose ranks differ in one bit.

   (c) (10%) Estimate the network diameter for odd k, assuming d is sufficiently larger than k.

   (d) (30%) Argue whether this network is superior or inferior to the hypercube.