Data Reorganization

1. In general, what is the purpose of data reorganization and of having a parallel pattern to do it?

2. In many physical system simulations, multiple data values might be stored together in a data structure for each cell in the system, representing certain physical properties. For instance, suppose we are simulating the atmosphere during 24 hours for a 10km x 10km x 10km volume where each cell is 10m x 10m x 10m in size (1M cells total) and contains data about velocity, temperature, and ozone content. The data structure might look something like:

   ```
   struct Cell {
       float velocity, temp, ozone;
   };
   Cell atmosphere[1000,1000,1000];
   ```

   Calculating the next values for each Cell can be done simultaneously for each field. That is, 1/3 of the threads could be used to update the velocity, 1/3 to update the temperature, and the remaining 1/3 to update the ozone content. (Note, each update probably will involve a stencil computation, but that is not the point of the problem.) Suppose we want to then execute all of the update in parallel.

   (a) Do you see any issues that might arise concerning data access?
   (b) What affects could there be on performance?
   (c) How would you resolve the issues using a parallel data reorganization pattern? Explain.

3. Do you see an opportunity for using a data reorganization pattern in preparing for a stencil computation? Explain.

Stencil Pattern

1. Imagine we’re working on data that can be represented in 3 dimensions as 1000 x 1000 x 1000 cells. We would like to perform a stencil operation on each cell. The stencil is a 3 x 3 x 3 cube around each cell (including the current cell) for a total of 18 cells per stencil operation. When partitioning this data among 8 threads, each thread will be responsible for a 500 x 500 x 500 chunk of data.
(a) Describe what the ghost cell regions will be like in this computation.

(b) Suppose we would like to use more threads in this computation. How many threads would we need if we wanted each "chunk" to be 250 x 250 x 250? 125 x 125 x 125? (Show work)

(c) Using your answers from the previous questions, describe how the halo regions change relative to the rest of the data. Consider the ratio of ghost cells to non-ghost cells. How does this change effect our computation?

2. Given a halo region in a stencil computation, why might we want to increase the "depth" of our halo region? Why might making this "depth" too large be a bad thing?

3. Consider the following nested for loops:

```plaintext
1 for i = 0; i < n; i++ do
  2     for j = 0; j < m; j++ do
    3       A[i][j] = f(A[i-1][j], A[i][j-1], A[i-1][j-1], B[i][j]);
  4     end
1 end
```

Despite the dependencies between loop iterations, this can still be parallelized. Describe how this operation would be parallelized. (tip: this is known as a recurrence pattern and is covered in the book)