Homework I

(50 points)

1. Suppose the best sequential algorithm for a problem requires $2n^2\log(\sqrt{n})$ steps for n data items. What is the maximum number of steps for a parallel algorithm to be *cost optimal* using n^2 processors? Is this ever possible?

2. Prove that there are no cyles of odd length in a hypercube of dimension d.

3. (a) How many nodes are there which are k distance away (considering only shortest routes) from a given node in a d-dimensional hypercube? Explain your result.

   (b) Using the formula above, show that average distance from a source processor to any other processor is d/2 where d is the dimension of the hypercube. Note that hypercube has a symmetric structure, therefore your result is true for any source processor.
   Hint: First, obtain the sum of all the distances from the source processor, and then divide that total value by the number of processors. Assume that the distance to itself is zero. You may also need to use the following relations:

$$C\begin{pmatrix} n \\ x \end{pmatrix} = \frac{n}{x}\, C\begin{pmatrix} n-1 \\ x-1 \end{pmatrix}$$

$$\sum_{i=0}^{n} C\begin{pmatrix} n \\ i \end{pmatrix} = 2^n$$

   where $C\begin{pmatrix} n \\ x \end{pmatrix}$ reads as "combination of n things taken x at a time".

4. In *one-to-all personalized communication*, a single processor sends a unique message of size m to every other processor.
   (i) Prove that the lower bound for the time complexity of this operation is $O(m * p)$ where p is the total number of processors in the system.
   (ii) Explain how you would perform *one-to-all personalized communication* on a hypercube multicomputer in $O(m * p)$ time. Draw figures if necessary.

5. Show all the steps required to sort the following sequence in increasing order in parallel using the

   (a) **bitonic sort** algorithm

   (b) **odd-even transposition sort** algorithm

   Show your steps in the form of a table as given below. What is the time complexity of each algorithm in terms of the input size N ?

```
          P0  P1  P2  P3  P4  P5  P6  P7  P8  P9  P10 P11 P12 P13 P14 P15
          4   7   3   19   8   6  22   5  10   2  27   9   1   18  13  12
 ------------------------------------------------------------------------
 STEP 1:
                .......                 .......                 .......

 Parallel Time Complexity =
```