CS546 Parallel Processing
Fall 1999, Dr. Xian-He Sun
Exam #1 (October 28, 1999)

Name:_____ (Last, First)

Score:_____ (100 pts.)

Make sure there are nine problems on five pages. Read each problem very carefully before answering it. Try to make your answers short and clear.

1. What is parallel processing? (5 pts.)

2. What is the difference between parallel processing and supercomputing? (5 pts.)

3. What does SISD, SIMD, and MIMD stand for? What is the difference between SPMD and SIMD? (10 pts.)

4. Define the Degree of Parallelism (5 pts.).

5. What does UMA and NUMA stand for? What is the major difference between NUMA machine and message-passing architecture? (10 pts.)

6. Give a brief answer for each of the questions:

   - Define *Speedup*
   - What is the difference between *fixed-size speed, fixed-time speedup*, and *memory-bounded speedup*?
   - Which two of the three speedups are *Scaled Speedup*? (15 pts.)

7. Consider the labels s and t of two processors A and B in a hypercube. The total number of bit positions at which these two labels differ is called the Hamming distance between them. Prove that the Hamming distance between A and B is the shortest distance between A and B in terms of communication links. (20 pts.)

8. Figure 1 gives a procedure for one-to-all broadcast of a message X from processor 0 to a d-dimensional hypercube. AND and XOR are bitwise logical-and and exclusive-or operations, respectively. Based on Figure 1, give a procedure for single-node accumulation on a d-dimensional hypercube. Each processor contributes a messages X containing m words, and processor 0 is the destination of the sum. What is the total time taken by your procedure on a p-processor hypercube with store-and-forward routing (15 pts.).

9. Given the sequential code

```
for i:=1 to 10 do
  b_i = 0
  for j:=1 to 10 do
    b_i = b_i + a_ij;
  endfor
endfor
```

Assume a parallel computer with 10 processors is available, write pseudocode (with a diagram if necessary) to show (15 pts.)

(a)) How the above sequential code can be parallelized via concurrent data parallel computation

(b)) How the above sequential code can be parallelized via pipelined data parallel computation