commodity clusters [88]. Feitelson et al. have written an overview of ParPar, "a general-purpose, multi-user, MPP-like-system, using only off-the-shelf components" [24].

If you want to assemble your own commodity cluster, *Beowulf Cluster Computing with Linux* by Sterling et al. is a practical guide to the construction, management, and programming of commodity PC clusters running the Linux operating system [105].

2.10 EXERCISES

2.1 Draw hypercube networks with two, four, and eight nodes. Make sure you label the nodes. Is a hypercube network with n nodes a subgraph of a hypercube network with $2n$ nodes?

2.2 How many different ways can a d-dimensional hypercube be labeled?

2.3 The distance between nodes u and v in a graph is the length of the shortest path from u to v. Given a d-dimensional hypercube and a designated source node s, how many nodes are distance i from s, where $0 \leq i \leq d$?

2.4 Prove that if node u is distance i from node v in a hypercube, then there are $i!$ different paths of length i from u to v (though some hypercube edges may appear in more than one path).

2.5 Prove that if node u is distance i from node v in a hypercube, then there are i paths of length i from u to v that share no edges.

2.6 Prove that a hypercube has no cycles of odd length.

2.7 Give an algorithm that routes a message from node u to node v in an n-node hypercube in no more than $\log n$ steps.

2.8 Draw shuffle-exchange networks with two, four, and eight nodes. Make sure you label the nodes. Is a shuffle-exchange network with n nodes a subgraph of a shuffle-exchange network with $2n$ nodes?

2.9 Given a shuffle-exchange network with 2^k nodes, under what circumstances are nodes u and v exactly $2k - 1$ link traversals apart?

2.10 Give an algorithm that routes a message from node u to node v in an n-node shuffle-exchange network in no more than $2 \log n - 1$ messages.

2.11 An **omega network** is an indirect topology based upon the perfect shuffle interconnection pattern [66]. Figure 2.23 illustrates an omega network for eight processors. Consider an omega network connecting $n = 2^k$ processors.

a. How many switching elements are in the network?

b. What is the diameter of the network?

c. What is the bisection width of the network?

d. What is the maximum number of edges per switching node?

e. Does the network have constant edge length as the number of nodes increases?

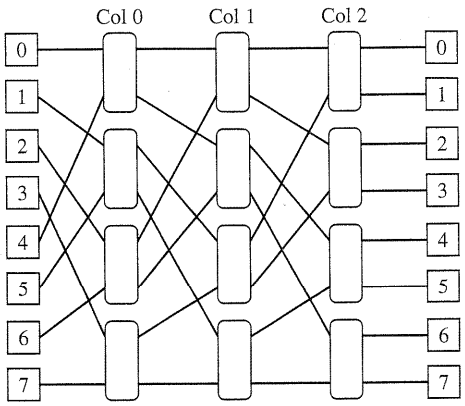

Figure 2.23 An omega network connecting eight processors, represented by squares.

2.12 Assume $n = 2^k$ processors are connected by an omega network (Figure 2.23). Design an algorithm to route a message from processor i to processor j. (Hint: Represent the destination address j as a binary number.)

2.13 Why are processor arrays well suited for executing data-parallel programs?

2.14 Given a processor array containing eight processing elements, each capable of performing 10 million integer operations per second, determine the performance in millions of operations per second of this processor array adding two integer vectors, for all vector sizes from 1 to 50.

2.15 Estimate the efficiency of a processor array executing a case statement with k cases. Assume all the instructions inside the case statement are parallel instructions, and assume all instructions take the same amount of time to execute.

a. What is the efficiency if each case contains the same number of instructions?

b. What is the efficiency if case i has I_i instructions and the probability of a processing element being active inside case i is P_i?

2.16 Why are large data and instruction caches desirable in multiprocessors?

2.17 Why is the number of processors in a centralized multiprocessor limited to a few dozen?

2.18 A directory-based protocol is a popular way to implement cache coherence on a distributed multiprocessor.

a. Why should the directory be distributed among the multiprocessor's local memories?

b. Why are the contents of the directory not replicated?

2.19 Continue the illustration of a directory-based cache coherence protocol begun in Figure 2.16. Assume the following five operations now occur in the order listed: CPU 2 reads X, CPU 2 write 5 to X, CPU 1 reads X, CPU 0 reads X, CPU 1 writes 9 to X. Show the states of the directories, caches, and memories after each of these operations.

2.20 Do some research and find, for each category in Flynn's taxonomy, at least one commercial computer fitting that category. (It is OK to name a computer that is no longer available, but you may not name a computer mentioned in this book.)

2.21 Continue the example of the operation of a systolic priority queue begun in Figure 2.22 by illustrating the states it would pass through as it processed these five requests: Insert 4, Extract Minimum, Insert 11, Insert 9, Extract Minimum.

2.22 Explain why contemporary supercomputers are invariably multicomputers.