

Parallel Computing Practice Exam

Generated on June 26, 2025

Difficulty: Medium

1. Describe a parallel algorithm for finding the connected components in an undirected graph using a message-passing paradigm. Explain how the algorithm handles the distribution of work across multiple processors and how it detects termination. Detail the communication patterns between the processors and discuss potential bottlenecks.
2. Implement a parallel merge sort algorithm using the fork-join model. Assume you have access to a shared memory system. Describe the steps involved in recursively dividing the data, sorting sub-arrays in parallel, and merging the sorted sub-arrays. Analyze the time complexity of your algorithm in terms of the number of processors and the input size.
3. Consider the problem of matrix multiplication of two large matrices A and B. Design a parallel algorithm using a distributed memory model where each processor is responsible for a sub-matrix. Outline the algorithm's steps, detailing how the processors communicate to exchange data and how the final result is assembled. Discuss the impact of communication overhead on the algorithm's performance.
4. Explain how you would adapt Dijkstra's algorithm for finding the shortest paths in a graph to a parallel environment. Assume a shared-memory model. Discuss different strategies for parallelizing the algorithm, considering both the computational load and the synchronization requirements. Compare the potential speedups achieved with different parallelization strategies.
5. You are given a large unsorted array of integers. Design a parallel algorithm to find the kth smallest element in this array using a randomized selection algorithm. Describe how you would partition the array into smaller sub-arrays and how you would recursively select the kth smallest element within each sub-array. Analyze the expected time complexity of your algorithm, taking into account the overhead of parallel operations.