Coral S. Schmidt Montilla #148830

Assessment 4.1

Dekker's Algorithm, Peterson's Algorithm, and Lamport's Bakery Algorithm are all software techniques that try to achieve mutual exclusion, which prevents many threads from simultaneously accessing shared resources. These algorithms ensure that only one thread enters the critical area at a time. They do, however, take different approaches and are more sophisticated. If both threads attempt to join their crucial regions at the same time, Dekker's Algorithm may jam up. Peterson's Algorithm overcomes this issue by providing a 'preferred thread' variable, although it still uses busy waiting, which might lead to performance inefficiencies. Lamport's Bakery Algorithm, on the other hand, uses a ticketing system suggestive of a bakery queue to ensure equal resource access. However, it may need more scalability in large-scale systems due to its centralized nature. While all three algorithms provide mutual exclusion, their suitability varies depending on the system requirements. Dekker's and Peterson's algorithms are more straightforward and may be more suitable for smaller systems. At the same time, Lamport's Bakery Algorithm offers fairness and scalability advantages, making it a better fit for distributed systems with multiple processors. Its centralized structure, however, may provide scalability issues in large-scale systems. While all three algorithms support mutual exclusion, their appropriateness differs according to the system's needs. Dekker and Peterson's algorithms are more straightforward and may be better suited to smaller systems. At the same time, Lamport's Bakery Algorithm provides fairness and scalability benefits, making it a better fit for distributed systems with several processors.