*Bài toán triết gia ăn ‘đêm’.*

Bài toán “Triết gia ăn tối” đề cập vấn đề chính yếu gì?

1. Bài toán “Triết gia ăn tối” đề cập vấn đề chính yếu gì?
   1. Tranh chấp các tài nguyên chia sẻ riêng biệt giữa những cặp tiến trình (race condition).
2. Bài toán “Triết gia ăn tối” nếu sử dụng Semaphore thì chúng được khởi tạo như thế nào?
   1. Semaphore chopstick[5], tất cả gán bằng 1.
3. Bài toán “Triết gia ăn tối” có thể được giải quyết bằng phương pháp nào để tránh bị tắt nghẽn (deadlock)?
   1. Bỏ quan sát (Monitor) và các lệnh test().

*Bài toán Bộ ghi – Bộ đọc (writer – reader).*

1. Bài toán “Bộ ghi – Bộ đọc” đề cập vấn đề chính yếu là gì?
   1. Dữ liệu chia sẻ mà chỉ có một vài tiến trình mới có nhu cầu cập nhật dữ liệu.
2. Bài toán “Bộ ghi – Bộ đọc” có đặc trưng gì?
   1. Chỉ có các bộ đọc mới có thể cập nhật dữ liệu chia sẻ.
3. Bài toán “Bộ ghi – Bộ đọc” các biến số được khởi tạo như thế nào?
   1. Semaphore rw\_mutex =1, mutex =1, read\_count=0;
4. Bài toán “Bộ ghi – Bộ đọc” có biến thể thứ 2, nó khác gì với ban đầu?
   1. Nếu một bộ ghi mới đến nó sẽ được thi nhanh nhất có thể.

*Bài toán Bộ đếm giới bạn (Bounded Buffer).*

1. Bài toán “Bộ đếm giới hạn” đề cập vấn đề chính yếu gì?
   1. Gửi và nhận gói tin qua bộ nhớ chia sẻ có kích thước nhất định.
2. Bài toán “Bộ đếm giới hạn” có thể giải quyết bằng bao nhiêu biến số Semaphore?
   1. Mutex, full, empty.