

Lab 06: Synchronization

Course: Operating Systems

Problem 1

Đây là một bài toán về xử lý đồng bộ, tranh chấp một tài nguyên dùng chung. Cụ thể tài khoản ngân hàng là tài nguyên được chia sẻ dùng chung cho 2 quá trình, ở đây là quá trình người chồng rút tiền và người vợ thêm tiền vào. Bài toán tương tự mô hình của bài toán producer - consumer.

Gọi tài khoản dùng chung là V, hành động rút tiền của người chồng là A, hành động thêm tiền vào của người vợ là B, khi đó các khả năng có thể xảy ra:

- + A, B xảy ra xen kẽ, hoặc liên tục không cùng lúc sử dụng V tại 1 thời điểm, ở đây các quá trình này diễn ra bình thường.

- + Khi A và B cùng lúc sử dụng V tại 1 thời điểm, xảy ra sự tranh chấp tài nguyên, dẫn đến sai sót, sai lệch hoặc lỗi trong quá trình thực hiện A, B, dẫn đến trạng thái tài khoản ngân hàng V có sự sai sót, có thể nhiều hơn hoặc ít hơn so với lý thuyết.

Để giải quyết tranh chấp trên, ta có thể sử dụng kỹ thuật mutex lock, hoặc semaphore để xử lý các quá trình đồng bộ trên.

Problem 2

So sánh về mặt performance của 2 cách ở lab 5 (sử dụng mảng để lưu trữ giá trị count ở mỗi thread) và ở lab 6 (sử dụng mutex lock xử lý quá trình đồng bộ dữ liệu chia sẻ giữa các thread):

- + Lab 5: đảm bảo không xảy ra tranh chấp dữ liệu dùng chung, đảm bảo tính chính xác của giải thuật, tốc độ xử lý cao. Tuy nhiên sử dụng nhiều vùng nhớ hơn so với lab 6.

- + Lab 6: Sử dụng ít vùng nhớ hơn cách lab 5, cụ thể chỉ sử dụng 1 biến count toàn cục thực hiện đếm, tuy nhiên cần sử dụng kỹ thuật mutex để đồng bộ dữ liệu chia sẻ, đảm bảo tính chính xác của giải thuật. Do đó thời gian thực thi so với lab 5 là lâu hơn do mỗi thread cần phải chờ cho đến khi biến chia sẻ được rỗi.