

Quy trình Đồng bộ hóa



Bài tập thực hành

6.1 Trong Phần 6.4, chúng tôi đã đề cập rằng việc tắt ngắt thường xuyên có thể ảnh hưởng đến đồng hồ của hệ thống. Giải thích tại sao nó có thể và làm thế nào những tác động đó có thể được giảm thiểu.

Trả lời: Đồng hồ hệ thống được cập nhật tại mỗi lần ngắt đồng hồ. Nếu ngắt bị vô hiệu hóa — đặc biệt là trong một khoảng thời gian dài — đồng hồ hệ thống có thể dễ dàng mất thời gian chính xác. Đồng hồ hệ thống cũng được sử dụng cho mục đích lập lịch trình. Ví dụ, lượng tử thời gian cho một quá trình được biểu thị bằng một số tích tắc đồng hồ. Tại mỗi lần ngắt đồng hồ, bộ lập lịch sẽ xác định xem lượng tử thời gian cho quá trình hiện đang chạy đã hết hạn hay chưa. Nếu ngắt đồng hồ bị vô hiệu hóa, bộ lập lịch không thể chỉ định chính xác lượng tử thời gian. Hiệu ứng này có thể được giảm thiểu bằng cách tắt ngắt đồng hồ trong khoảng thời gian rất ngắn.

6.2 *Vấn đề người hút thuốc lá.* Hãy xem xét một hệ thống có ba người hút thuốc quy trình và một đại lý quy trình. Mỗi người hút liên tục cuộn một điếu rồi hút. Nhưng để cuộn và hút một điếu thuốc, người hút cần 3 nguyên liệu: thuốc lá, giấy và diêm. Một trong các quy trình của người hút thuốc có giấy, quy trình khác có thuốc lá, và quy trình thứ ba có diêm. Đại lý có nguồn cung cấp vô hạn của cả ba nguyên liệu. Người đại diện đặt hai trong số các thành phần trên bàn. Người hút thuốc có thành phần còn lại sau đó chế biến và hút một điếu thuốc, báo hiệu tác nhân đã hoàn thành. Sau đó, đại lý đưa ra hai trong số ba thành phần khác, và chu trình lặp lại. Viết chương trình để đồng bộ hóa tác nhân và người hút thuốc bằng cách sử dụng đồng bộ hóa Java.

Trả lời: Vui lòng tham khảo trang Web hỗ trợ để biết giải pháp mã nguồn.

6,3 Đưa ra lý do tại sao Solaris, Windows XP và Linux triển khai nhiều cơ chế khóa. Mô tả các trường hợp hợp mà họ

sử dụng spinlocks, mutexes, semaphores, mutex thích ứng và các biến điều kiện. Trong mỗi trường hợp, hãy giải thích lý do tại sao cơ chế đó là cần thiết. **Trả lời:** Các hệ điều hành này cung cấp các cơ chế khóa khác nhau tùy thuộc vào nhu cầu của nhà phát triển ứng dụng. Spinlocks hữu ích cho các hệ thống đa xử lý trong đó một luồng có thể chạy trong một vòng lặp bận (trong một khoảng thời gian ngắn) thay vì phải chịu chi phí đưa vào hàng đợi ngủ. Mutexes rất hữu ích để khóa tài nguyên. Solaris 2 sử dụng mutex thích ứng, có nghĩa là mutex được thực hiện với một khóa quay trên các máy đa xử lý. Semaphores và biến điều kiện là những công cụ thích hợp hơn để đồng bộ hóa khi tài nguyên phải được lưu giữ trong thời gian dài, vì kéo sợi không hiệu quả trong thời gian dài.

6.4 Giải thích sự khác biệt, về mặt chi phí, giữa ba loại lưu trữ dễ bay hơi, không dễ bay hơi và ổn định.

Trả lời: Lưu trữ đa năng đề cập đến bộ nhớ chính và bộ nhớ đệm và rất nhanh. Tuy nhiên, lưu trữ dễ bay hơi không thể tồn tại khi hệ thống gặp sự cố hoặc tắt nguồn hệ thống. Lưu trữ không bay hơi vẫn tồn tại sự cố hệ thống và hệ thống bị sập nguồn. Đĩa và băng là những ví dụ về lưu trữ không bay hơi. Gần đây, các thiết bị USB sử dụng bộ nhớ chỉ đọc chương trình có thể xóa được (EPROM) đã xuất hiện cung cấp khả năng lưu trữ không linh hoạt. Lưu trữ ổn định đề cập đến lưu trữ về mặt kỹ thuật có thể *không bao giờ* bị mất vì có các bản sao lưu dự phòng của dữ liệu (thường là trên đĩa).

6.5 Giải thích mục đích của cơ chế **điểm kiểm tra**. Các điểm kiểm tra nên được thực hiện bao lâu một lần? Mô tả tần suất của các điểm kiểm tra ảnh hưởng như thế nào:

- Hiệu suất hệ thống khi không xảy ra lỗi
- Thời gian cần để khôi phục sau sự cố hệ thống
- Thời gian cần để khôi phục sau sự cố đĩa

Trả lời: Bản ghi nhật ký điểm kiểm tra chỉ ra rằng bản ghi nhật ký và dữ liệu sửa đổi của nó đã được ghi vào bộ nhớ ổn định và giao dịch không cần phải thực hiện lại trong trường hợp hệ thống gặp sự cố. Rõ ràng, các trạm kiểm soát càng được thực hiện thường xuyên thì càng ít có khả năng các bản cập nhật thừa sẽ phải được thực hiện trong quá trình khôi phục.

- Hiệu suất hệ thống khi không xảy ra lỗi — Nếu không có lỗi nào xảy ra, hệ thống phải chịu chi phí thực hiện các trạm kiểm soát về cơ bản là không cần thiết. Trong tình huống này, việc thực hiện các trạm kiểm soát ít thường xuyên hơn sẽ dẫn đến hiệu suất hệ thống tốt hơn.
- Thời gian cần để khôi phục sau sự cố hệ thống — Sự tồn tại của bản ghi điểm kiểm tra có nghĩa là một thao tác sẽ không phải thực hiện lại trong quá trình khôi phục hệ thống. Trong tình huống này, các trạm kiểm soát được thực hiện càng thường xuyên thì thời gian khôi phục sau sự cố hệ thống càng nhanh.
- Thời gian cần để khôi phục sau sự cố đĩa — Sự tồn tại của bản ghi điểm kiểm tra có nghĩa là một thao tác sẽ không phải thực hiện lại trong quá trình khôi phục hệ thống. Trong tình huống này, thường xuyên hơn

các trạm kiểm soát được thực hiện, thời gian khôi phục càng nhanh sau sự cố đĩa.

6.6 Giải thích khái niệm nguyên tử giao dịch.

Trả lời: Một giao dịch là một chuỗi các hoạt động đọc và ghi dựa trên một số dữ liệu, theo sau là một hoạt động cam kết. Nếu không thể hoàn thành chuỗi hoạt động trong một giao dịch, thì giao dịch đó phải được hủy bỏ và các hoạt động đã diễn ra phải được khôi phục lại. Điều quan trọng là chuỗi hoạt động trong một giao dịch xuất hiện dưới dạng một hoạt động không thể phân chia để đảm bảo tính toàn vẹn của dữ liệu được cập nhật. Nếu không, dữ liệu có thể bị xâm phạm nếu các hoạt động từ hai (hoặc nhiều) giao dịch khác nhau được trộn lẫn với nhau.

6.7 Cho thấy rằng một số lịch trình có thể thực hiện được theo giao thức khóa hai pha nhưng không thể thực hiện được theo giao thức dấu thời gian và ngược lại. **Trả lời:** Lịch biểu được phép trong giao thức khóa hai pha nhưng không được phép trong giao thức dấu thời gian là:

bước chân	T_0	T_1	Quyền ưu tiên
1	lock-S ($MỘT$)		
2	đọc ($MỘT$)		
3		khóa-X (B)	
4		viết (B)	
5		mở khóa (B)	
6	lock-S (B)		
7	đọc (B)		$T_1 \rightarrow T_0$
số 8	mở khóa ($MỘT$)		
9	mở khóa (B)		

Lịch biểu này không được phép trong giao thức dấu thời gian vì ở bước 7, dấu thời gian W của B là 1.

Lịch biểu được phép trong giao thức dấu thời gian nhưng không được phép trong giao thức khóa hai pha là:

bước chân	T_0	T_1	T_2
1	viết ($MỘT$)		
2		viết ($MỘT$)	
3			viết ($MỘT$)
4	viết (B)		
5		viết (B)	

Lịch trình này không thể có thêm hướng dẫn khóa để làm cho nó hợp pháp theo giao thức khóa hai giai đoạn vì T_1 phải mở khóa ($MỘT$) giữa các bước 2 và 3 và phải khóa (B) giữa các bước 4 và 5.

6.8 Các Chờ đã() trong tất cả các ví dụ chương trình Java là một phần của trong khi vòng.

Giải thích lý do tại sao bạn luôn cần sử dụng trong khi tuyên bố khi sử dụng Chờ đã() và tại sao bạn không bao giờ sử dụng nếu tuyên bố. **Trả lời:** Đây là một vấn đề quan trọng cần nhấn mạnh! Java chỉ cung cấp thông báo ẩn danh — bạn không thể thông báo cho một chuỗi nhất định rằng một chứng chỉ

20 Chương 6 Đồng bộ hóa quy trình

điều kiện tain là đúng. Khi một luồng được thông báo, nó có trách nhiệm kiểm tra lại điều kiện mà nó đang chờ. Nếu một chuỗi không kiểm tra lại điều kiện, nó có thể đã nhận được thông báo mà không đáp ứng điều kiện.