

Lập lịch CPU



Bài tập thực hành

5.1 Thuật toán lập lịch CPU xác định thứ tự thực hiện các quy trình đã lên lịch của nó. Được cho các quy trình được lập lịch trên một bộ xử lý, có bao nhiêu lịch trình khác nhau có thể có? Đưa ra một công thức về n .

Trả lời: $n!$ (ngai thừa $= n \times (n-1) \times (n-2) \times \dots \times 2 \times 1$).

5.2 Xác định sự khác biệt giữa lập lịch trước và không đặt trước.

Trả lời: Lập lịch trước cho phép một quá trình bị gián đoạn khi đang thực thi, lấy CPU đi và phân bổ nó cho một quá trình khác. Lập lịch không tự do đảm bảo rằng một quá trình chỉ từ bỏ quyền kiểm soát CPU khi nó kết thúc với sự bùng nổ CPU hiện tại của nó.

5.3 Giả sử rằng các quy trình sau đến để thực thi vào thời gian được chỉ định. Mỗi quá trình sẽ chạy trong khoảng thời gian được liệt kê. Khi trả lời các câu hỏi, hãy sử dụng lập lịch trình không phóng thích và dựa trên tất cả các quyết định dựa trên thông tin bạn có tại thời điểm phải đưa ra quyết định.

Proces S	MỘTrrival Tim e	Blần đầu tiên
P_1	0,0	số 8
P_2	0,4	4
P_3	1,0	1

Một. Thời gian quay vòng trung bình cho các quy trình này với thuật toán lập lịch FCFS là bao nhiêu?

- b. Thời gian quay vòng trung bình cho các quy trình này với thuật toán lập lịch SJF là bao nhiêu?
- c. Thuật toán SJF được cho là sẽ cải thiện hiệu suất, nhưng lưu ý rằng chúng tôi đã chọn chạy quy trình P_1 tại thời điểm 0 vì chúng tôi không biết rằng hai quy trình ngắn hơn sẽ sớm đến. Tính toán thời gian quay vòng trung bình sẽ là bao nhiêu nếu CPU không hoạt động trong 1 đơn vị đầu tiên và sau đó lập lịch SJF được sử dụng. Hãy nhớ rằng các quy trình P_1 và P_2 đang chờ đợi trong thời gian nhàn rỗi này, vì vậy thời gian chờ đợi của họ có thể tăng lên. Thuật toán này có thể được gọi là lập lịch tri thức trong tương lai.

Trả lời:

- Một. 10,53
- b. 9,53
- c. 6,86

Hãy nhớ rằng thời gian quay vòng là thời gian kết thúc trừ thời gian đến, vì vậy bạn phải trừ thời gian đến để tính thời gian quay vòng. FCFS là 11 nếu bạn quên trừ thời gian đến.

5.4 Lợi thế nào khi có các kích thước lượng tử thời gian khác nhau ở các cấp độ khác nhau của hệ thống xếp hàng đa cấp?

Trả lời: Các quy trình cần được bảo dưỡng thường xuyên hơn, chẳng hạn như các quy trình tương tác như trình chỉnh sửa, có thể nằm trong hàng đợi với lượng tử thời gian nhỏ. Các quy trình không cần bảo dưỡng thường xuyên có thể nằm trong hàng đợi với lượng tử lớn hơn, yêu cầu ít công tắc ngữ cảnh hơn để hoàn thành quá trình xử lý và do đó sử dụng máy tính hiệu quả hơn.

5.5 Nhiều thuật toán lập lịch trình CPU được tham số hóa. Ví dụ, thuật toán RR yêu cầu một tham số để chỉ ra lát cắt thời gian. Hàng đợi phản hồi đa cấp yêu cầu các tham số để xác định số lượng hàng đợi, các thuật toán lập lịch cho mỗi hàng đợi, các tiêu chí được sử dụng để di chuyển các quy trình giữa các hàng đợi, v.v.

Do đó, những thuật toán này thực sự là tập hợp các thuật toán (ví dụ, tập hợp các thuật toán RR cho mọi lát cắt thời gian, v.v.). Một bộ thuật toán có thể bao gồm một bộ thuật toán khác (ví dụ, thuật toán FCFS là thuật toán RR với lượng tử thời gian vô hạn). Mỗi quan hệ nào (nếu có) giữ giữa các cặp tập hợp thuật toán sau đây?

- Một. Ưu tiên và SJF
- b. Hàng đợi phản hồi đa cấp và FCFS
- c. Ưu tiên và FCFS
- d. RR và SJF

Trả lời:

- Một. Công việc ngắn nhất có mức độ ưu tiên cao nhất.
- b. Cấp thấp nhất của MLFQ là FCFS.

C. FCFS dành ưu tiên cao nhất cho công việc đã tồn tại lâu nhất.

d. Không có.

5,6 Giả sử rằng một thuật toán lập lịch (ở cấp độ lập lịch CPU ngắn hạn) ủng hộ những quá trình đã sử dụng ít thời gian xử lý nhất trong quá khứ gần đây. Tại sao thuật toán này lại ưu tiên các chương trình ràng buộc I / O nhưng lại không bỏ đói các chương trình ràng buộc CPU vĩnh viễn?

Trả lời: Nó sẽ ưu tiên các chương trình ràng buộc I / O vì yêu cầu cụm CPU tương đối ngắn của chúng; tuy nhiên, các chương trình ràng buộc CPU sẽ không chết đói vì các chương trình ràng buộc I / O sẽ tương đối thường xuyên rời bỏ CPU để thực hiện I / O của chúng.

5,7 Phân biệt giữa lập lịch PCS và SCS.

Trả lời: Lập lịch PCS được thực hiện cục bộ cho quá trình. Đó là cách thư viện luồng lên lịch cho các luồng trên các LWP có sẵn. Lập lịch SCS là tình huống hệ điều hành lập lịch cho các luồng nhân. Trên các hệ thống sử dụng nhiều-một hoặc nhiều-nhiều, hai mô hình lập kế hoạch về cơ bản là khác nhau. Trên các hệ thống sử dụng một-một, PCS và SCS đều giống nhau.

5,8 Giả sử một hệ điều hành ánh xạ các luồng cấp người dùng tới hạt nhân bằng cách sử dụng mô hình nhiều-nhiều, nơi ánh xạ được thực hiện thông qua việc sử dụng LWP. Hơn nữa, hệ thống cho phép các nhà phát triển chương trình tạo các luồng thời gian thực. Có cần thiết phải liên kết một chuỗi thời gian thực với một LWP không? **Trả lời:** Có, nếu không thì một luồng người dùng có thể phải cạnh tranh để giành được một LWP có sẵn trước khi được lên lịch thực sự. Bằng cách liên kết luồng người dùng với một LWP, không có độ trễ khi chờ đợi một LWP có sẵn; luồng người dùng thời gian thực có thể được lên lịch ngay lập tức.

