



Ôn tập các dạng bài Hệ điều hành NEU

Hệ điều hành (Đại học Kinh tế Quốc dân)



Scan to open on Studocu

Ôn tập các dạng bài Hệ điều hành NEU

➔ Đề các năm không thay đổi về dạng bài mấy

Nội dung:

| | |
|----------|---|
| Chương 2 | 1. Fork() 2. FCFS, SJF, SRJF, RR, Priority, Multi Level |
| Chương 3 | 3. First fit, Best fit, Worst fit 4. FIFO, LRU, OPT |
| Chương 4 | 5. FCFS, SSTF, SCAN, C-SCAN, LOOK, C-LOOK 6. Slide 11 Thầy Lâm |
| Chương 5 | 7. Chmod – slide 38 8. I/O |
| Chương 7 | 9. Ma trận truy cập |

Tài liệu SITDE mới: [Phương pháp giải bài tập Hệ điều hành.pdf](#)

BT chương 2: Quản lý tiến trình

- Lập lịch tiến trình: [HỆ ĐIỀU HÀNH PTIT | Bài toán Điều độ \(youtube.com\)](#). Chú ý round robin họ làm ko chuẩn, đề nghị tự tìm video riêng để xem
- Thuật toán banker: [Bài tập hệ điều hành - Deadlock 2 \(thuật toán Banker - giải thuật nhà băng\) \(youtube.com\)](#)., [Bài tập hệ điều hành - Deadlock 1, có Request \(thuật toán Banker - giải thuật nhà băng\) \(youtube.com\)](#) Lưu ý khác biệt ở đây là Thay vì quay lại cấp phát, thì ta sẽ đi hết vòng này tới vòng kia (tùy theo cách trường dạy - trường mình là đi hết vòng này mới tới vòng kia). xem xong thực hành bài trên stide

BT chương 3: Quản lý bộ nhớ

- Tìm số bit của bộ nhớ: [HỆ ĐIỀU HÀNH PTIT | Bài toán Phân chương và Phân trang bộ nhớ \(youtube.com\)](#)
- Phân chương động: best-fit, worst-fit, first-fit: [\(40\) CẤP PHÁT BỘ NHỚ TRONG HỆ ĐIỀU HÀNH PHÂN VÙNG ĐỘNG - YouTube](#)
- (Check var độ đúng video) Nạp trang: [HỆ ĐIỀU HÀNH PTIT | Bài toán Đổi trang \(youtube.com\)](#) - không học clock; học lru, opt, fcfs thôi. Ở opt thì họ chọn tùy thích, nhưng mình học theo nếu chọn thì kick cái thẳng dùng lâu nhất ở phần tử cuối
- Dạng TLBs với hit-ratio, tính thời gian truy xuất bộ nhớ trong hệ thống: <https://stackoverflow.com/questions/18550370/calculate-the-effective-access-time>
- Tính toán hiện tượng phân mảnh nội khi sử dụng hệ thống phân trang
- Số bytes 1 tiến trình cần / Kích thước trang = x

→ 1 Tiến trình cần tối thiểu x (làm tròn lên) trang

- Trang x phân mảnh nội với số bytes dư thừa là:
Kích thước trang - (Số bytes 1 TT cần MOD kích thước trang) → Hoàn thành ý a
- (ý b) Nếu đề hỏi dung lượng bộ nhớ lãng phí bởi phân mảnh nội
= Tổng số tiến trình x số byte dư thừa phân mảnh nội

BT chương 4: Quản lý hệ thống file

- Phân quyền file, chmod: siêu dễ doc sitde là được và tài liệu khá đủ ở đây: <https://viblo.asia/p/phan-quyen-trong-linux-yMnKMbDNZ7P>
- Các phương pháp phân phối vùng lưu trữ: Phân phối liên tục, Phân phối liên kết, Phân phối chỉ mục → Hoang mang

Giả sử một file có 100 khối (blocks). Cần thực hiện bao nhiêu thao tác vào ra (IO Operations) để ghi thêm 1 khối hoặc xóa 1 khối tại các vị trí:

- Đầu file
- Sau khối 50
- Cuối file

Giải sử:

- Trong phương phân phối liên tục, đầu của khối không thể thêm (vì khối trước khối đầu tiên đã được phân phối cho 1 file khác)
- Trong phương phân phối chỉ mục, khối chỉ mục (index) đã có sẵn trong bộ nhớ

Kiến thức cơ bản:

Phân bố liên tục là sơ đồ phân bổ đơn giản nhất. Ý tưởng là lưu trữ mỗi tệp dưới dạng chạy liên tục của không gian đĩa. Ví dụ: nếu một tệp có kích thước 20-KB, chúng ta sẽ phân bổ 20 khối liên tiếp cho tệp đó.

Cấp phát theo liên kết là một phương pháp để lưu trữ tệp trong đó tệp được giữ dưới dạng danh sách đĩa được liên kết các khối. Từ đầu tiên của mỗi khối được sử dụng như một con trỏ đến từ tiếp theo và phần còn lại của khối được sử dụng cho dữ liệu.

512= 508(data)+4 (index)

Cấp phát theo chỉ mục là một phương thức liên kết với mỗi tệp một cấu trúc dữ liệu được gọi là nút chỉ mục trong đó liệt kê các thuộc tính và địa chỉ đĩa của các khối tệp. Với nút chỉ mục, nó là có thể tìm thấy tất cả các khối của tệp.

| | Liên tục | Liên kết | Chỉ mục |
|-------------------------|---|--|---|
| Thêm 1 khối ở đầu | 200+1=201 200(đọc vào bộ nhớ và ghi từ bộ nhớ ra 100 khối, dịch về sau)+ 1(ghi khối mới) | 1 thay đổi con trỏ trong khối điều khiển tệp (FCB) để trỏ đến khối mới trong đĩa. | 1 (index block) Chỉ cần ghi khối mới và cập nhật chỉ mục trong bộ nhớ. |
| Thêm 1 khối sau khối 50 | 100+1=101 100(đọc vào bộ nhớ và ghi từ bộ nhớ ra 50 khối sau khối 50)+1 (ghi khối mới) | 50+1+1=52 -Đọc 50 khối đầu tìm để đọc khối 51 -Sao chép liên kết của khối 50 vào khối mới (trong bộ nhớ chính)-51 cũ 52 mới - 1 cập nhật liên kết của khối 50 | 1 Chỉ cần ghi khối mới và cập nhật chỉ mục trong bộ nhớ. |

| | | | |
|--------------------|---|---|-------------------------|
| | | 101 sau đó cập nhật con trỏ khối cuối cùng trong bộ nhớ. | |
| Xóa khối đầu tiên | -198 (đọc 99 khối còn lại + dịch 99 khối còn lại để trở thành 99 khối do trống khối đầu đã bị xóa). Cách này tối ưu về mặt quản lý bộ nhớ vì sẽ tiết kiệm được vị trí của khối đầu tiên khi đã bị xóa. - 0 (với cách này thì chỉ cần cập nhật trong bộ nhớ) (tuy nhiên nếu các em trả lời 1 trong 2 đáp án là 0 hoặc 198 thì đều đúng) | 1 Đọc trong khối đầu tiên để lấy con trỏ của khối thứ hai và sau đó đặt con trỏ khối đầu tiên trỏ đến khối thứ hai. | 0 (chỉ cập nhật bộ nhớ) |
| Xóa khối 50 | 100(đọc và ghi 50 khối sau khối 50) (thao tác dịch các khối từ 51-100) | 50+1 = 51 -Đọc 50 khối đầu tiên vào trong bộ nhớ để lấy thông tin con trỏ khối 51 -1 (sử dụng pointer khối 49 sẽ kết nối với khối 51) | 0 (chỉ cập nhật bộ nhớ) |
| Xóa khối cuối cùng | 0(chỉ cần cập nhật là đã trống) | 99+1=100 99: Đọc 99 đầu tiên 1 sử dụng pointer khối 99 thành -1(null) | 0 (chỉ cập nhật bộ nhớ) |

BT chương 5: quản lý nhập xuất io

- Cylinder quản lý hệ thống vào ra fcfs, sstf, scan, c-scan, look, c-look: [\[PDF\]Hệ Điều Hành - Đh Bách Khoa Hn - Ts Phạm Đăng Hải \(cuuduongthancong.com\)](#) . Tải chương 5 về đọc slide siêu hay: từ 38 đến hết và [\[KMA\] Chap8_MassStorageManager \[NGUYỄN LÝ HỀ ĐIỀU HÀNH \]](#)
- Tính toán thời gian truy cập trung bình: làm bài này rất lưu ý MB Mb Gb GB và đơn vị thời gian (s) và (ms) (cẩn thận trên slide làm sai bài này trong ví dụ)

$$\text{Thời gian trễ trung bình} = \frac{1}{2} \text{ Thời gian trễ} = \frac{1}{2} \times \frac{60}{\text{số vòng/phút}} (s)$$

$$\text{TG truy cập trễ} = \text{TG truy cập trung bình} = \text{TG tìm kiếm trung bình} + \text{TG trễ trung bình}$$

$$\text{TG I/O trung bình} = \text{TG truy cập trung bình} + \frac{\text{Lượng truyền vào}}{\text{Tốc độ truyền}} + \text{TG xử lý trên đầu đọc)$$

BT chương 7: về ma trận quyền truy xuất / ma trận quyền truy cập

- đọc file GS_07 ở file thầy gửi trong mục sách hệ điều hành. Đọc phần cần đọc thôi, khó hiểu thì tự tìm trang khác mà đọc. (slide làm như cc, ko cần tham khảo)

Copy: nếu một quyền truy xuất R trong $access[i,j]$ được đánh dấu là R^* thì có thể sao chép nó sang một phần tử $access[k,j]$ khác (mở rộng quyền truy xuất R trên cùng đối tượng O_j nhưng trong miền bảo vệ D_k).

Transfer : nếu một quyền truy xuất R trong $access[i,j]$ được đánh dấu là R^+ thì có thể chuyển nó sang một phần tử $access[k,j]$ khác (chuyển quyền truy xuất R^+ trên đối tượng O_j sang miền bảo vệ D_k).

Owner : nếu $access[i,j]$ chứa quyền truy xuất $owner$ thì tiến trình hoạt động trong miền bảo vệ D_i có thể thêm hoặc xóa các quyền truy xuất trong bất kỳ phần tử nào trên **cột j** (có quyền thêm hay bớt các quyền truy xuất trên đối tượng O_j trong những miền bảo vệ khác).

Control : nếu $access[i,j]$ chứa quyền truy xuất $control$ thì tiến trình hoạt động trong miền bảo vệ D_i có thể xóa bất kỳ quyền truy xuất nào trong các phần tử trên **dòng j** (có quyền bỏ bớt các quyền truy xuất trong miền bảo vệ D_j).

| | | |
|---------------|----------------------|----------------------|
| object | F₁ | F₂ |
|---------------|----------------------|----------------------|

Miền này Chuyển sang miền kia → Miền này có thể thực hiện quyền của miền kia