







>>> COURSE MATERIAL <<<

INTRODUCTION TO OPERATING SYSTEM / Course ID 502047

Hướng dẫn tự học, chỉ sử dụng để tham khảo và có thể thay đổi tùy vào từng học kỳ/nhóm lớp.

	Programming Exercises	-----4
	Exam multichoice questions	12345--89---3-5-78901--4----9---3--67-----
	Discussion questions	-----67---12---6-----2--567--0---45--89-----
	Advance discussions	-----0---4-----3---8--12-----0123-

BÀI TẬP CHƯƠNG 10 – BỘ NHỚ ẢO

10.1 Trong trường hợp nào xảy ra lỗi trang? Mô tả các hành động được thực hiện bởi hệ điều hành khi xảy ra lỗi trang.

10.2 Giả sử rằng máy tính có chuỗi tham chiếu trang cho một quy trình có m khung (ban đầu tất cả đều trống). Chuỗi tham chiếu trang có độ dài p và n số trang riêng biệt được truy cập. Trả lời những câu hỏi này cho từng thuật toán thay thế trang:

- Giới hạn dưới của số lượng lỗi trang là gì?
- Giới hạn trên của số lỗi trang là gì?

10.3 Xem xét các thuật toán thay thế trang sau đây. Xếp hạng các thuật toán này theo thang điểm năm từ Rất kém đến mức Hoàn hảo, theo tỷ lệ lỗi trang của chúng. Những thuật toán nào có xảy ra bất thường của Belady?

- Thay thế LRU

b. Thay thế FIFO

c. Thay thế tối ưu

d. Thay thế cơ hội thứ hai

10.4 Một hệ điều hành hỗ trợ bộ nhớ ảo phân trang. Bộ vi xử lý có thời gian chu kỳ là 1 micro giây. Mất thêm 1 micro giây để truy cập một trang khác với trang hiện tại. Các trang có 1.000 từ và thiết bị phân trang là một trống quay với tốc độ 3.000 vòng / phút và chuyển 1 triệu từ mỗi giây. Các phép đo thống kê sau đây được lấy từ hệ thống:

- Một phần trăm của tất cả các lệnh được thực hiện đã truy cập một trang khác với trang hiện tại.
- Trong số các lệnh truy cập một trang khác, 80 phần trăm truy cập một trang đã có trong bộ nhớ.
- Khi cần một trang mới, trang bị thay thế đã được sửa đổi 50 phần trăm thời gian.

Tính thời gian thực hiện lệnh hiệu quả trên hệ thống này, giả sử rằng hệ thống chỉ chạy một tiến trình và bộ xử lý không hoạt động trong khi chuyển trống.

10.5 Xem xét bảng phân trang cho một hệ thống có địa chỉ ảo và vật lý cùng có độ dài 12 bit và các trang 256 byte.

Page	Page Frame
0	–
1	2
2	C
3	A
4	–
5	4
6	3
7	–
8	B
9	0

Danh sách các khung trang sẵn sàng là D, E, F (nghĩa là D đứng đầu danh sách, E đứng thứ hai và F là cuối cùng). Dấu gạch ngang “–” cho biết trang không nằm trong bộ nhớ.

Chuyển đổi các địa chỉ ảo sau thành địa chỉ vật lý tương đương của chúng theo hệ thập lục phân. Tất cả các số được đưa ra trong thập lục phân.

- 0x9EF
- 0x111
- 0x700
- 0x0FF

10.6 Thảo luận về các chức năng phân cứng cần thiết để hỗ trợ phân trang theo yêu cầu [demand paging].

10.7 Xét mảng hai chiều A:

```
int A [ ] [ ] = new int [100] [100];
```

Trong đó A [0] [0] ở vị trí 200 trong hệ thống bộ nhớ phân trang với các trang có kích thước 200. Một tiến trình nhỏ nhân ma trận nằm trong trang 0 (vị trí 0 đến 199). Do đó, mọi lệnh nạp vào sẽ từ trang 0.

Đối với ba khung trang, có bao nhiêu lỗi trang được tạo bởi các vòng lặp khởi tạo mảng sau đây? Sử dụng thay thế LRU và giả sử rằng khung trang 1 chứa tiến trình và hai trang còn lại ban đầu trống.

```
a. for(int j = 0; j <100; j ++)  
    for(int i = 0; i <100; i ++)  
        A [i] [j] = 0;  
b. for (int i = 0; i <100; i ++)  
    for(int j = 0; j <100; j ++)  
        A [i] [j] = 0;
```

10.8 Xem xét chuỗi tham chiếu trang sau:

1, 2, 3, 4, 2, 1, 5, 6, 2, 1, 2, 3, 7, 6, 3, 2, 1, 2, 3, 6.

Có bao nhiêu lỗi trang sẽ xảy ra đối với các thuật toán thay thế sau, giả sử hệ thống có 1, 2, 3, 4, 5, 6, 7 khung? Hãy nhớ rằng tất cả các khung trang ban đầu đều trống, vì vậy các trang truy cập lần đầu tiên đều bị lỗi trang.

- Thay thế LRU
- Thay thế FIFO
- Thay thế tối ưu

10.9 Xem xét chuỗi tham chiếu trang sau:

7, 2, 3, 1, 2, 5, 3, 4, 6, 7, 7, 1, 0, 5, 4, 6, 2, 3, 0, 1.

Giả sử hệ thống phân trang theo yêu cầu với ba khung trang, có bao nhiêu lỗi trang sẽ xảy ra đối với các thuật toán thay thế sau đây?

- Thay thế LRU
- Thay thế FIFO
- Thay thế tối ưu

10.10 Giả sử rằng bạn muốn sử dụng thuật toán phân trang yêu cầu với bit tham chiếu (chẳng hạn như mô hình thay thế cơ hội thứ hai hoặc mô hình tập làm việc working-set), nhưng phần cứng không cung cấp một thuật toán cần thiết. Phác thảo cách bạn có thể mô phỏng một bit tham chiếu ngay cả khi phần cứng không được cung cấp bởi phần cứng hoặc giải thích lý do tại sao không thể làm như vậy. Nếu có thể, hãy tính toán chi phí sẽ là bao nhiêu.

10.11 Bạn hãy nghĩ ra một thuật toán thay thế trang mới mà bạn nghĩ có thể là tối ưu. Trong một số trường hợp thử nghiệm méo mó, dị thường Belady đã xảy ra.

Giải thuật mới này có tối ưu? Giải thích câu trả lời của bạn.

10.12 Phân đoạn tương tự như phân trang nhưng sử dụng các “trang” có kích thước thay đổi. Hãy đề xuất hai thuật toán thay thế phân đoạn, một thuật toán dựa trên sơ đồ thay thế trang FIFO và thuật toán còn lại dựa trên sơ đồ thay thế trang LRU. Hãy nhớ rằng vì các phân đoạn không có cùng kích thước, phân đoạn được chọn để thay thế có thể quá nhỏ để có đủ các vị trí liên tiếp cho phân đoạn cần thiết. Xem xét các chiến lược cho các hệ thống mà các phân đoạn không thể di dời và các chiến lược cho các hệ thống có thể di dời.

10.13 Hãy xem xét một hệ thống máy tính theo yêu cầu trong đó mức độ đa lập trình hiện đang cố định là 4. Hệ thống gần đây đã được đo để xác định việc sử dụng CPU và đĩa phân trang. Ba kết quả thay thế được hiển thị dưới đây. Đối với mỗi trường hợp, những khả năng gì đang xảy ra?

Có thể tăng mức độ đa chương trình để tăng mức sử dụng CPU không? Và kỹ thuật phân trang có giúp ích trong tình huống đó không?

- a. Sử dụng CPU 13 phần trăm; sử dụng đĩa 97 phần trăm
- b. Sử dụng CPU 87 phần trăm; sử dụng đĩa 3 phần trăm
- c. Sử dụng CPU 13 phần trăm; sử dụng đĩa 3 phần trăm

10.14 Một hệ điều hành cho một máy sử dụng các thanh ghi cơ sở và giới hạn, nhưng được sửa đổi để cung cấp bảng phân trang. Bảng trang có thể được thiết lập để mô phỏng các thanh ghi cơ sở và giới hạn không? Làm thế nào để hiện thực được, hoặc tại sao nó không thể hiện thực được?

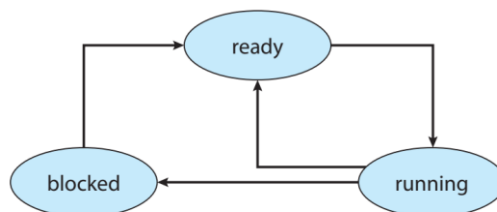
10.15 Giả sử rằng một chương trình vừa tham chiếu một địa chỉ trong bộ nhớ ảo. Mô tả một kịch bản trong đó mỗi điều sau đây có thể xảy ra. (Nếu không thể tồn tại kịch bản như, hãy giải thích tại sao.)

- TLB bị trượt và không có lỗi trang
- Lỗi TLB với lỗi trang
- TLB trùng và không có lỗi trang
- TLB bị lỗi trang

10.16 Một cách nhìn đơn giản hóa các trạng thái tiểu trình có trạng thái: sẵn sàng, đang chạy và bị chặn. Trong đó một tiểu trình sẵn sàng đang chờ để được lên lịch, tiểu trình đang chạy trên bộ xử lý hoặc bị chặn (ví dụ, chờ I/O).

Giả sử một tiểu trình đang ở trạng thái chạy, trả lời các câu hỏi sau và giải thích câu trả lời của bạn:

- a. Tiểu trình có thay đổi trạng thái nếu nó phát sinh lỗi trang hay không? Nếu vậy, nó sẽ thay đổi thành trạng thái nào?
- b. Tiểu trình có thay đổi trạng thái nếu nó bị trượt TLB nhưng đã được phân giải trong bảng phân trang hay không? Nếu vậy, nó sẽ thay đổi thành trạng thái nào?
- c. Tiểu trình có thay đổi trạng thái nếu tham chiếu địa chỉ được phân giải trong bảng trang hay không? Nếu vậy, nó sẽ thay đổi thành trạng thái nào?



10.17 Xem xét một hệ thống sử dụng phân trang theo yêu cầu thuần túy [pure demand paging].

- a. Khi một tiến trình đầu tiên bắt đầu thực thi, tỷ lệ lỗi trang sẽ như thế nào?
- b. Khi tập làm việc cho một tiến trình được tải vào bộ nhớ, tỷ lệ lỗi trang sẽ như thế nào?

c. Giả sử rằng một tiến trình bị di chuyển đi nơi khác trong bộ nhớ (tính địa phương bị thay đổi) và kích thước của tập làm việc mới quá lớn để được lưu trữ trong bộ nhớ trống có sẵn.

Chỉ ra một số khả năng mà các nhà thiết kế hệ thống có thể chọn để xử lý tình huống này.

10.18 Sau đây là bảng phân trang cho một hệ thống có địa chỉ vật lý và địa chỉ ảo đều là 12 bit với trang có kích cỡ 256 byte. Các khung trang sẵn sàng sẽ được phân bổ theo thứ tự 9, F, D. Dấu gạch ngang tại một khung trang cho biết trang đang tham chiếu không nằm trong bộ nhớ.

Page	Page Frame
0	0 x 4
1	0 x B
2	0 x A
3	–
4	–
5	0 x 2
6	–
7	0 x 0
8	0 x C
9	0 x 1

Chuyển đổi các địa chỉ ảo sau thành địa chỉ vật lý tương đương của chúng theo hệ thập lục phân. Tất cả các số được đưa ra trong thập lục phân. Trong trường hợp có lỗi trang, bạn phải sử dụng một trong các khung trang sẵn sàng để cập nhật bảng phân trang và tiếp tục phân giải địa chỉ logic thành địa chỉ vật lý tương ứng.

- 0x2A1
- 0x4E6
- 0x94A
- 0x316

10.19 Tính năng sao-chép-khi-ghi [copy-on-write] là gì và trong trường hợp nào thì sử dụng tính năng này có lợi? Cần hỗ trợ phần cứng nào để thực hiện tính năng này?

10.20 Một máy tính cung cấp cho người dùng không gian bộ nhớ ảo kích cỡ 2^{32} byte. Máy tính này có 2^{22} byte bộ nhớ vật lý. Bộ nhớ ảo được triển khai bằng cách phân trang và kích thước trang là 4.096 byte.

Một quá trình người dùng tạo địa chỉ ảo 11123456. Giải thích cách hệ thống thiết lập vị trí vật lý tương ứng. Phân biệt giữa hoạt động phần mềm và phần cứng.

10.21 Giả sử rằng chúng ta có bộ nhớ phân trang theo yêu cầu. Bảng phân trang được tổ chức trong các thanh ghi. Phải mất 8 mili giây để xử lý lỗi trang nếu có khung trống hoặc nếu trang bị thay thế không có thay đổi

(không cần phải swap out) và 20 mili giây nếu trang bị thay thế có thay đổi. Thời gian truy cập bộ nhớ là 100 nano giây.

Giả sử rằng trang cần thay thế có thay đổi chiếm 70%. Tỷ lệ lỗi trang tối đa chấp nhận được trong thời gian truy cập hiệu quả không quá 200 nano giây là bao nhiêu?

10.22 Xem xét bảng phân trang cho một hệ thống có địa chỉ vật lý và địa chỉ ảo cùng là 16 bit và trang có kích cỡ 4.096 byte.

Page	Page Frame	Reference Bit
0	9	0
1	–	0
2	10	0
3	15	0
4	6	0
5	13	0
6	8	0
7	12	0
8	7	0
9	–	0
10	5	0
11	4	0
12	1	0
13	0	0
14	–	0
15	2	0

Bit “reference” cho một trang được đặt thành 1 khi trang được tham chiếu. Định kỳ, một tiểu trình đặt lại giá trị 0 cho tất cả các bit “reference”. Dấu gạch ngang trong khung trang cho biết trang không nằm trong bộ nhớ.

Thuật toán thay thế trang là LRU cục bộ và tất cả các số được cung cấp ở dạng thập phân.

a. Chuyển đổi các địa chỉ ảo sau (theo hệ thập lục phân) thành các địa chỉ vật lý tương đương. Bạn có thể cung cấp câu trả lời ở dạng thập lục phân hoặc thập phân. Đồng thời thiết lập giá trị bit “reference” cho mục nhập thích hợp trong bảng phân trang.

- 0x621C

- 0xF0A3

- 0xBC1A

- 0x5BAA

- 0x0BA1

b. Sử dụng các địa chỉ trên làm ví dụ và chỉ ra địa chỉ logic (theo hệ thập lục phân) đã dẫn đến lỗi trang.

c. Các khung trang nào mà thuật toán thay thế trang LRU sẽ chọn trong việc giải quyết lỗi trang?

10.23 Khi xảy ra lỗi trang, tiến trình yêu cầu trang phải bị chặn [blocking] trong khi chờ trang được đưa từ đĩa vào bộ nhớ vật lý. Giả sử rằng tồn tại một tiến trình với các tiểu trình mức người dùng và việc ánh xạ các tiểu trình người dùng đến các tiểu trình nhân là nhiều-đến-một. Nếu một tiểu trình người dùng phát sinh lỗi trang trong khi truy cập ngăn xếp của nó, thì các tiểu trình người dùng khác thuộc cùng tiến trình cũng sẽ bị ảnh hưởng bởi lỗi trang đó hay không?, có phải chúng cũng phải đợi trang bị lỗi được đưa vào bộ nhớ hay không? Giải thích chi tiết.

10.24 Áp dụng thuật toán thay thế (1) FIFO, (2) LRU và (3) tối ưu (OPT) cho các chuỗi tham chiếu trang sau:

- 2, 6, 9, 2, 4, 2, 1, 7, 3, 0, 5, 2, 1, 2, 9, 5, 7, 3, 8, 5

- 0, 6, 3, 0, 2, 6, 3, 5, 2, 4, 1, 3, 0, 6, 1, 4, 2, 3, 5, 7

- 3, 1, 4, 2, 5, 4, 1, 3, 5, 2, 0, 1, 1, 0, 2, 3, 4, 5, 0, 1

- 4, 2, 1, 7, 9, 8, 3, 5, 2, 6, 8, 1, 0, 7, 2, 4, 1, 3, 5, 8

- 0, 1, 2, 3, 4, 4, 3, 2, 1, 0, 0, 1, 2, 3, 4, 4, 3, 2, 1, 0

Cho biết số lượng lỗi trang cho mỗi thuật toán trong trường hợp phân trang theo yêu cầu với 3 khung trang.

10.25 Giả sử rằng bạn đang theo dõi tốc độ mà con trỏ trong thuật toán đồng hồ di chuyển. (Con trỏ chỉ ra trang nạn nhân để thay thế.) Có thể nói gì về hệ thống nếu xuất hiện từng dấu hiệu sau:

a. Con trỏ đang di chuyển nhanh.

b. Con trỏ đang di chuyển chậm.

10.26 Thảo luận về các tình huống trong đó thuật toán thay thế trang ít được sử dụng thường xuyên nhất (LFU) tạo ra ít lỗi trang hơn thuật toán thay thế trang được sử dụng gần đây nhất (LRU). Ngược lại, trong những trường hợp nào thì LRU ít lỗi trang hơn LFU.

10.27 Thảo luận về các tình huống trong đó thuật toán thay thế trang được sử dụng thường xuyên nhất (MFU) tạo ra ít lỗi trang hơn thuật toán thay thế trang được sử dụng gần đây nhất (LRU). Ngược lại, trong những trường hợp nào thì LRU ít lỗi trang hơn MFU.

10.28 Hệ điều hành KHIE (đọc là “h-hi”) sử dụng thuật toán thay thế FIFO cho các trang lưu trú và nhóm khung trang sẵn sàng của các trang được sử dụng gần đây. Giả sử rằng nhóm khung trang sẵn sàng này được quản lý bằng chính sách thay thế LRU. Trả lời các câu hỏi sau:

a. Nếu xảy ra lỗi trang và trang không tồn tại trong nhóm khung trang sẵn sàng, làm thế nào để không gian trống được tạo ra cho trang mới được yêu cầu?

b. Nếu xảy ra lỗi trang và trang tồn tại trong nhóm khung trang sẵn sàng, làm thế nào trang được thiết lập và nhóm khung trang sẵn sàng được quản lý để tạo khoảng trống cho trang được yêu cầu?

c. Hệ thống suy thoái đến mức nào nếu số lượng trang lưu trú được thiết lập với số lượng là 1?

d. Hệ thống suy thoái đến mức nào nếu số lượng trang trong nhóm khung trang sẵn sàng bằng 0?

10.29 Xem xét hệ thống phân trang theo yêu cầu với các thông số sử dụng được đo theo thời gian sau:

Sử dụng CPU 20%

Đĩa cứng phục vụ phân trang 97,7%

Các thiết bị I / O khác 5%

Đối với mỗi hành động sau đây, hãy cho biết liệu nó sẽ (hoặc có khả năng) cải thiện việc sử dụng CPU hay không? Giải thích câu trả lời của bạn.

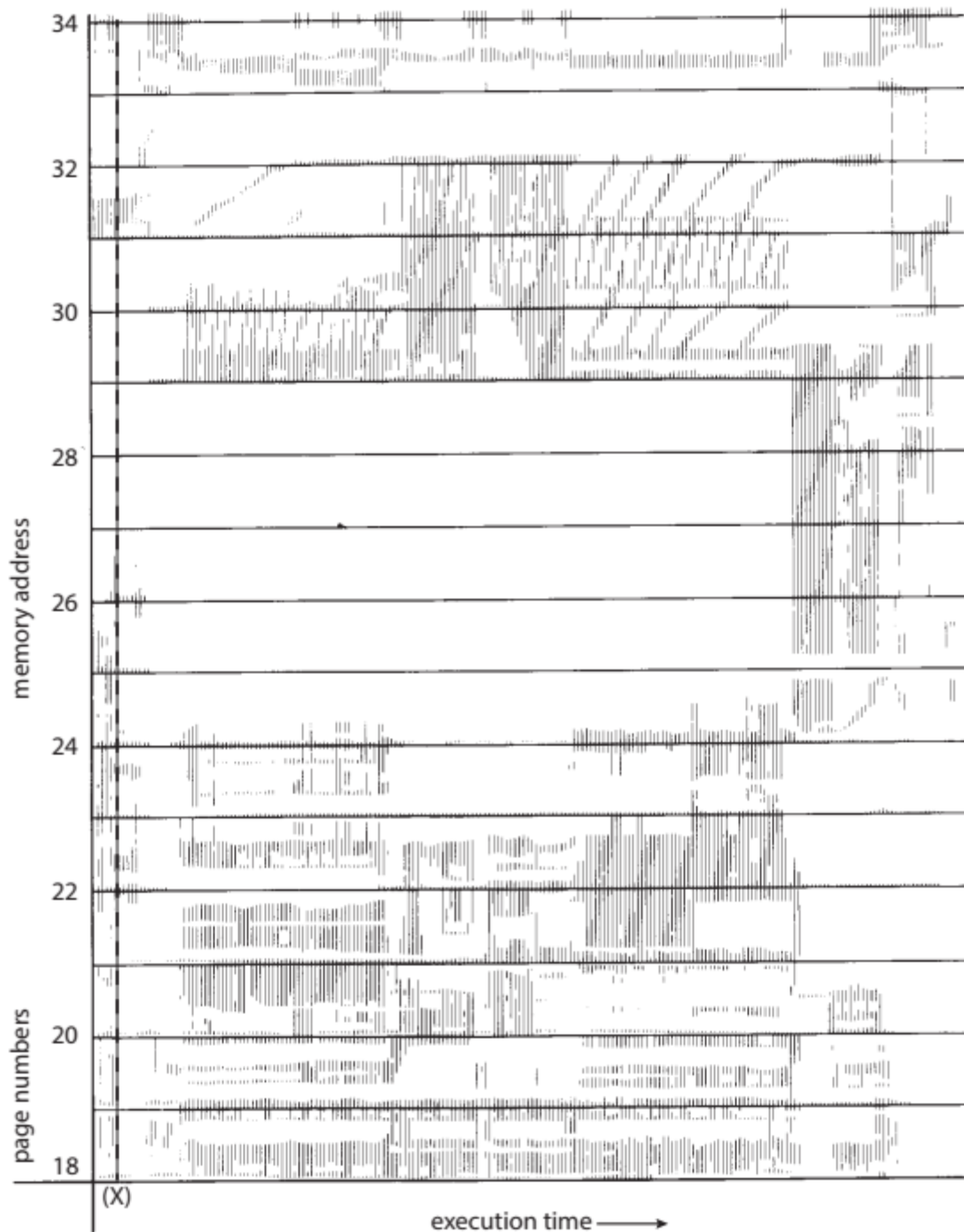
- a. Nâng cấp CPU nhanh hơn.
- b. Cài đặt một phân vùng đĩa cứng phục vụ phân trang lớn hơn.
- c. Tăng mức độ đa chương của hệ thống.
- d. Giảm mức độ đa chương của hệ thống.
- e. Cài đặt thêm dung lượng bộ nhớ chính.
- f. Cài đặt đĩa cứng nhanh hơn hoặc sử dụng RAID 0 (disk stripping).
- g. Thêm trang chuẩn bị cho các thuật toán tìm nạp trang.
- h. Tăng kích thước trang.

10.30 Giải thích tại sao các lỗi trang kích thước nhỏ mất ít thời gian để giải quyết hơn các lỗi trang kích thước lớn.

10.31 Giải thích tại sao bộ nhớ nén [compressed memory] được sử dụng trong các hệ điều hành cho thiết bị di động.

10.32 Giả sử rằng một máy tính cung cấp các lệnh có thể truy cập các vị trí bộ nhớ bằng cách sử dụng sơ đồ địa chỉ gián tiếp một cấp [one-level indirect addressing]. Trình tự lỗi trang nào phát sinh khi tất cả các trang của chương trình hiện không hoạt động và lệnh đầu tiên của chương trình là hoạt động tải bộ nhớ gián tiếp? Điều gì xảy ra khi hệ điều hành đang sử dụng kỹ thuật cấp khung theo từng tiến trình và chỉ có hai trang được phân bổ cho tiến trình này?

10.33 Xem xét các tham khảo trang sau đây:



Những trang nào đại diện cho địa phương tại thời điểm (X)?

10.34 Giả sử rằng chính sách thay thế của một hệ thống phân trang là kiểm tra từng trang thường xuyên và loại bỏ trang đó nếu nó không được sử dụng kể từ lần kiểm tra trước. Hệ thống sẽ đạt được gì và sẽ mất gì khi sử dụng chính sách này thay vì LRU hoặc thay thế cơ hội thứ hai?

10.35 Thuật toán thay thế trang tốt sẽ giảm thiểu số lượng lỗi trang. Chúng ta có thể đạt được sự giảm thiểu này bằng cách phân phối các trang được sử dụng nhiều đều trên tất cả bộ nhớ, thay vì để chúng cạnh tranh nhau trên một số lượng nhỏ các khung trang. Chúng ta có thể liên kết với mỗi khung trang một bộ đếm số lượng trang được liên kết với khung trang đó. Sau đó, để thay thế một trang, chúng ta có thể tìm kiếm khung trang bằng bộ đếm nhỏ nhất.

a. Xác định thuật toán thay thế trang bằng ý tưởng cơ bản này. Cụ thể, hãy giải quyết những vấn đề này:

- Giá trị ban đầu của các bộ đếm là bao nhiêu?
- Khi nào các bộ đếm tăng giá trị?
- Khi nào các bộ đếm giảm giá trị?
- Làm thế nào để chọn trang nạn nhân?

b. Có bao nhiêu lỗi trang xảy ra đối với thuật toán đã đề xuất cho chuỗi tham chiếu sau với 4 khung trang?

1, 2, 3, 4, 5, 3, 4, 1, 6, 7, 8, 7, 8, 9, 7, 8, 9, 5, 4, 5, 4, 2.

c. Số lỗi trang tối thiểu cho chiến lược thay thế trang tối ưu cho chuỗi tham chiếu trong phần b với 4 khung trang là bao nhiêu?

10.36 Xem xét hệ thống phân trang theo yêu cầu với đĩa phân trang có tổng thời gian truy cập và truyền dữ liệu trung bình là 20 mili giây. Địa chỉ được dịch thông qua một bảng phân trang trong bộ nhớ chính, với thời gian truy cập là 1 micro giây cho mỗi lần truy cập bộ nhớ. Do đó, mỗi tham khảo bộ nhớ thông qua bảng phân trang cần hai lần truy cập. Để cải thiện thời gian này, người ta đã thêm một bộ nhớ kết hợp [associative memory] giúp giảm thời gian truy cập cho một tham khảo bộ nhớ nếu chỉ mục cần tìm của bảng phân trang nằm trong bộ nhớ kết hợp.

Giả sử rằng 80 phần trăm số lượt truy cập nằm trong bộ nhớ kết hợp và trong số lượt truy cập còn lại, 10 phần trăm (hoặc 2 phần trăm tổng số) gây ra lỗi trang. Thời gian truy cập bộ nhớ hiệu EAT quả là bao nhiêu?

10.37 Nguyên nhân gây ra sự cố thua đậm [thrashing] là gì? Làm thế nào để hệ thống phát hiện một sự cố như vậy? Một khi nó phát hiện ra sự cố, hệ thống có thể làm gì để loại bỏ vấn đề này?

10.38 Có thể cho một tiến trình có hai tập làm việc [working set] hay không, một tập cho dữ liệu và một tập cho mã nguồn? Giải thích.

10.39 Xem xét tham số Δ được sử dụng để định nghĩa tập làm việc trong mô hình tập làm việc. Khi Δ được đặt giá trị thấp, tần suất lỗi trang và số lượng tiến trình hoạt động (không bị treo) hiện đang thực thi trong hệ thống sẽ bị ảnh hưởng như thế nào? Hiệu ứng khi Δ được đặt giá trị rất cao là gì?

10.40 Trong phân đoạn 1.024 KB, bộ nhớ được cấp phát bằng hệ thống bạn bè [buddy system].

Sử dụng hình 10.26 làm hướng dẫn, vẽ cây minh họa cách phân bổ các yêu cầu bộ nhớ sau:

- Yêu cầu 5-KB
- Yêu cầu 135 KB.
- Yêu cầu 14 KB.
- Yêu cầu 3 KB.
- Yêu cầu 12 KB.

Tiếp theo, sửa đổi cây cho các bản phát hành bộ nhớ sau. Thực hiện liên kết bất cứ khi nào có thể:

- Giải phóng 3 KB.
- Giải phóng 5 KB.
- Giải phóng 14 KB.
- Giải phóng 12 KB.

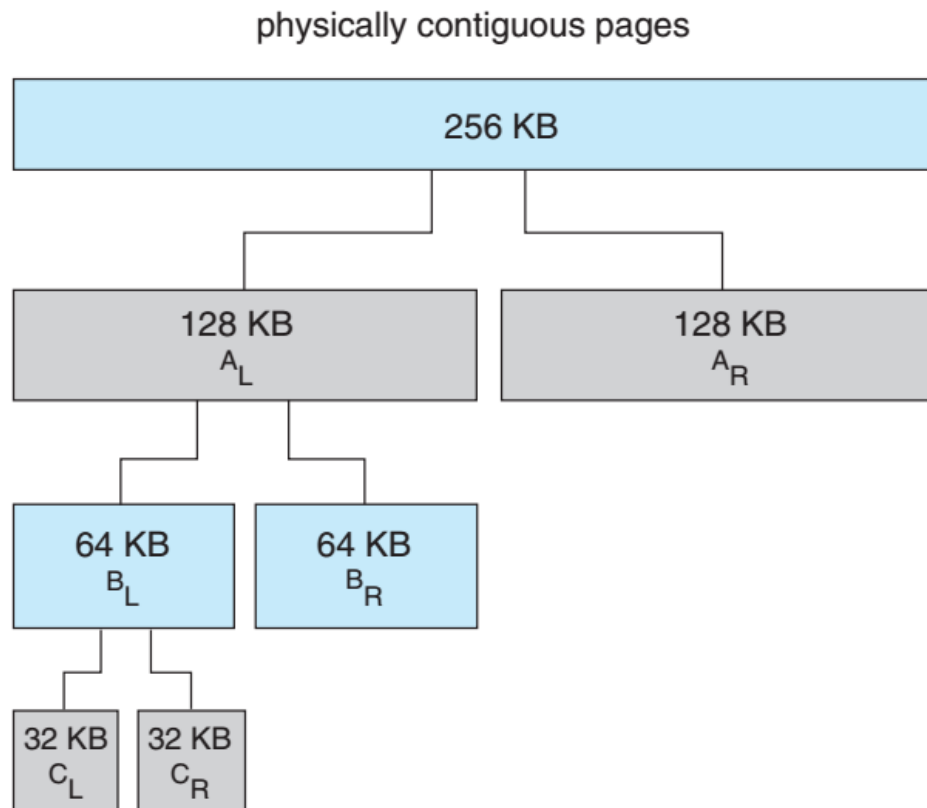


Figure 10.26 Buddy system allocation.

10.41 Một hệ thống cung cấp hỗ trợ cho các tiểu trình mức người dùng và mức kernel. Ánh xạ trong hệ thống này là một-một (có một tiểu trình nhân tương ứng cho mỗi tiểu trình người dùng). Liệu một tiến trình đa luồng bao gồm (a) một tập làm việc cho toàn bộ tiến trình hay (b) một tập làm việc cho mỗi luồng? Giải thích

10.42 Thuật toán cấp phát bản mỏng [slab-allocation] sử dụng bộ nhớ cache riêng cho từng loại đối tượng khác nhau. Giả sử có một bộ nhớ cache cho mỗi loại đối tượng, giải thích lý do tại sao mô hình này không triển khai tốt với hệ thống nhiều CPU. Những gì có thể được thực hiện để triển khai mở rộng đó hiệu quả hơn?

10.43 Xem xét một hệ thống cấp phát các trang có kích thước khác nhau cho các các tiến trình. Những lợi thế của một mô hình phân trang như vậy là gì? Những thay đổi nào cho hệ thống bộ nhớ ảo sẽ là cần thiết để cung cấp chức năng này?

10.44 Viết chương trình thực hiện các thuật toán thay thế trang FIFO, LRU và tối ưu (OPT) được trình bày trong Chương 10.4. Chương trình ban đầu cần tạo ra một chuỗi tham chiếu trang ngẫu nhiên trong đó số trang nằm trong khoảng từ 0 đến 9. Áp dụng chuỗi tham chiếu trang ngẫu nhiên cho mỗi thuật toán và ghi lại số lỗi trang phát sinh theo từng thuật toán. Truyền số lượng khung trang cho chương trình khi khởi động.