Chapter 7

# Quản lý bộ nhớ là gì? Các yêu cầu đối với quản lý bộ nhớ là gì?

* Quản lý bộ nhớ là công việc của hệ điều hành với sự hỗ trợ của phần cứng nhằm phân phối, sắp xếp các process trong bộ nhớ sao cho hiệu quả.
* Yêu cầu:
  + Cấp phát bộ nhớ cho các process
  + Tái định vị (relocation): khi swapping, …
  + Bảo vệ: phải kiểm tra truy xuất bộ nhớ có hợp lệ không
  + Chia sẻ: cho phép các process chia sẻ vùng nhớ chung
  + Kết gán địa chỉ nhớ luận lý của user vào địa chỉ thực

# Trình bày đặc điểm các loại địa chỉ bộ nhớ?

* Địa chỉ vật lý (physical address) (địa chỉ thực) là một vị trí thực trong bộ nhớ chính.
* Địa chỉ luận lý (logical address) là một vị trí nhớ được diễn tả trong một chương trình (còn gọi là địa chỉ ảo virtual address).

# Địa chỉ lệnh và dữ liệu được chuyển đổi thành địa chỉ thực có thể xảy ra tại những thời điểm nào?

* Compile time: nếu biết trước địa chỉ bộ nhớ của chương trình thì có thể kết gán địa chỉ tuyệt đối lúc biên dịch.
* Load time: vào thời điểm loading, loader phải chuyển đổi địa chỉ khả tái định vị thành địa chỉ thực dựa trên một địa chỉ nền.
* Excution time: khi trong quá trình thực thi, process có thể được di chuyển từ segment này sang segment khác trong bộ nhớ thì quá trình chuyển đổi địa chỉ được trì hoãn đến thời điểm thực thi.

# Liên kết động (dynamic linking) là gì?

* Quá trình link đến một module ngoài (external module) được thực hiện sau khi đã tạo xong load module (i.e file có thể thực thi, executable).

# Phân mảnh ngoại và phân mảnh nội là gì?

* Phân mảnh ngoại (external fragmentation)
  + Kích thước không gian nhớ còn trống đủ để thỏa mãn một yêu cầu cấp phát, tuy nhiên không gian nhớ này không liên tục ⇒ có thể dùng cơ chế kết khối (compaction) để gom lại thành vùng nhớ liên tục.
* Phân mảnh nội (internal fragmentation)
  + Kích thước vùng nhớ được cấp phát có thể hơi lớn hơn vùng nhớ yêu cầu.
    - Ví dụ: cấp một khoảng trống 18,464 bytes cho một process yêu cầu 18,462 bytes.
  + Hiện tượng phân mảnh nội thường xảy ra khi bộ nhớ thực được chia thành các khối kích thước cố định (fixed-sized block) và các process được cấp phát theo đơn vị khối. Ví dụ: cơ chế phân trang (paging).

# Như thế nào là phân vùng động, phân vùng cố định? Các chiến lược placement dùng để làm gì? Có những chiến lược nào?

* Phân vùng động:
  + Số lượng partition không cố định và partition có thể có kích thước khác nhau
  + Mỗi process được cấp phát chính xác dung lượng bộ nhớ cần thiết
  + Gây ra hiện tượng phân mảnh ngoại
* Phân vùng cố định:
  + Khi khởi động hệ thống, bộ nhớ chính được chia thành nhiều phần rời nhau gọi là các partition có kích thước bằng nhau hoặc khác nhau
  + Process nào có kích thước nhỏ hơn hoặc bằng kích thước partition thì có thể được nạp vào partition đó.
  + Nếu chương trình có kích thước lớn hơn partition thì phải dùng cơ chế overlay.
* Các chiến lược placement:
  + Dùng để quyết định cấp phát khối bộ nhớ trống nào cho một process
  + Mục tiêu: giảm chi phí compaction
  + Có 4 chiến lược:
    - Best-fit: chọn khối nhớ trống nhỏ nhất
    - First-fit: chọn khối nhớ trống phù hợp đầu tiên kể từ đầu bộ nhớ
    - Next-fit: chọn khối nhớ trống phù hợp đầu tiên kể từ vị trí cấp phát cuối cùng
    - Worst-fit: chọn khối nhớ trống lớn nhất

# Cơ chế phân trang là gì? Cơ chế chuyển đổi địa chỉ trong phân trang được thực hiện như thế nào?

* Cơ chế phân trang:
  + Bộ nhớ vật lý → khung trang (frame).
  + Bộ nhớ luận lý (logical memory) hay không gian địa chỉ luận lý là tập mọi địa chỉ luận lý mà một chương trình bất kỳ có thể sinh ra → page.
  + Bảng phân trang (page table) để ánh xạ địa chỉ luận lý thành địa chỉ thực.
* Chuyển đổi địa chỉ:
  + Địa chỉ luận lý gồm có:
    - Số hiệu trang (Page number) p
    - Địa chỉ tương đối trong trang (Page offset) d
  + Nếu kích thước của không gian địa chỉ ảo là 2m, và kích thước của trang là 2n (đơn vị là byte hay word tùy theo kiến trúc máy) thì: Bảng trang sẽ có tổng cộng 2m/2n = 2m - n mục (entry)

# Bảng trang được cài đặt và tổ chức như thế nào? Thời gian truy xuất hiệu dụng (effective access time) được xác định như thế nào?

* Bảng phân trang thường được lưu giữ trong bộ nhớ chính. Thường dùng một bộ phận cache phần cứng có tốc độ truy xuất và tìm kiếm cao, gọi là thanh ghi kết hợp (associative register) hoặc translation look-aside buffers (TLBs).
* Thời gian truy xuất hiệu dụng
* EAT = (ε + x)α + (ε + 2x)(1 – α)
* = (2 – α)x + ε

# Cơ chế hoán vị là gì?

* Một process có thể tạm thời bị swap ra khỏi bộ nhớ chính và lưu trên một hệ thống lưu trữ phụ. Sau đó, process có thể được nạp lại vào bộ nhớ để tiếp tục quá trình thực thi.

# (Bài tập mẫu)

# (Bài tập mẫu)

# (Bài tập mẫu)

# Một máy tính 32-bit địa chỉ, sử dụng một bảng trang 2 cấp. Địa chỉ ảo được phân bổ như sau: 9-bit dành cho bảng trang cấp 1, 11 bit cho bảng trang cấp 2 và còn lại cho offset. Cho biết kích thước một trang trong hệ thống và địa chỉ ảo có bao nhiêu trang?

# Giả sử địa chỉ ảo 32-bit được phân tách thành 4 trường a, b, c, d. 3 trường đầu tiên được dùng cho bảng trang 3 cấp, trường thứ 4 dành cho offset. Số lượng trang có phụ thuộc vào kích thước của cả 4 trường này không? Nếu không, những trường nào ảnh hưởng đến số lượng trang, những trường nào không ảnh hưởng?

# (Bài tập mẫu)

# Xét một hệ thống sử dụng kỹ thuật phân trang, với bảng trang được lưu trữ trong bộ nhớ chính.

a. Nếu thời gian cho một lần truy xuất bộ nhớ bình thường là 124 nanoseconds, thì mất bao nhiêu thời gian cho một thao tác truy xuất bộ nhớ trong hệ thống này?

b. Nếu sử dụng TLBs với hit-ratio (tỉ lệ tìm thấy) là 95%, thời gian để tìm trong TLBs bằng 34ns, tính thời gian cho một thao tác truy xuất bộ nhớ trong hệ thống (effective memory reference time).

# Xét một hệ thống sử dụng kỹ thuật phân trang, với bảng trang được lưu trữ trong bộ nhớ chính. Nếu sử dụng TLBs với hit-ratio (tỉ lệ tìm thấy) là 87%, thời gian để tìm trong TLBs là 24 ns. Thời gian truy xuất bộ nhớ trong hệ thống (effective memory reference time) là 175ns. Tính thời gian cho một lần truy xuất bộ nhớ bình thường?

# Biết thời gian truy xuất trong bộ nhớ thường không sử dụng TLBs là 250ns. Thời gian tìm kiếm trong bảng TLBs là 26ns. Hỏi xác suất tìm thấy trong TLBs bằng bao nhiêu nếu thời gian truy xuất trong bộ nhớ chính là 182ns?