HỆ THỐNG QUẢN LÝ BỘ NHỚ

1 Giới thiệu

1.1 Bộ Nhớ

Trong máy tính có nhiều loại bộ nhớ, trước mắt là 2 loại lớn : bộ nhớ trong (còn gọi là bộ nhớ chính) & bộ nhớ ngoài (bộ nhớ phụ /bộ nhớ thứ cấp). Ranh giới "trong" và "ngoài" ở đây không phải là vỏ máy mà là Bo mạch chính – nơi kết nối các thành phần quan trọng nhất của máy tính với nhau (HDD /SSD tuy nằm trong vỏ máy nhưng vẫn là bộ nhớ ngoài).

Bộ nhớ ngoài tương đối đa dạng: HDD, SSD, USB disk, Flash disk, Flash card, Optical disc (DVD, VCD,…), bộ nhớ của 1 máy khác có kết nối với máy đang xét (có thể bao gồm cả phần bộ nhớ trong)

Bộ nhớ trong bao gồm các loại: RAM, ROM, Cache, Register.

1.2 Quản lý Bộ Nhớ

Việc quản lý điều hành bộ nhớ ngoài cần tổ chức thành hai phần: cấp thấp (giả lập thành các thiết bị logic, hỗ trợ đọc /ghi trực tiếp các khối dữ liệu) và cấp cao (xây dựng và hỗ trợ truy xuất tập tin) sẽ được đảm nhiệm bởi hai thành phần quan trọng của OS: Hệ Thống Quản Lý Nhập Xuất và Hệ Thống Quản Lý Tập Tin.

Việc quản lý điều hành các thành phần bộ nhớ trong ROM, Cache, Register thì rất đơn giản và không không cần đến OS (của cả hệ thống máy tính). Riêng RAM thì khá phức tạp và rất quan trọng, nếu không có sự quản lý điều hành chính xác và khéo léo thì sẽ khó mà chạy được các chương trình trên máy tính – do đó cần một thành phần quan trọng của OS đảm nhiệm: Hệ Thống Quản Lý Bộ Nhớ (Memory System Management)

1.3 Hệ thống Quản lý Bộ Nhớ

Bộ Nhớ kể từ đây được mặc định là bộ nhớ chính (RAM). Đây là thiết bị lưu trữ duy nhất mà thông qua nó CPU mới có thể trao đổi thông tin với mội trường bên ngoài. RAM có dung lượng tuy không nhỏ (thường là vài đến vài chục GiB trên các máy cá nhân) nhưng không đủ lớn để lưu trữ dữ liệu và chương trình – đồng thời khi không còn nguồn điện nuôi thì nội dung trên RAM sẽ mất. Do đó chương trình & dữ liệu sẽ nằm trên bộ nhớ ngoài và khi cần chạy / truy xuất thì bắt buộc phải được nạp vào RAM.

Với hệ thống đơn chương (mỗi thời điểm chỉ cho phép một chương trình chạy), trong bộ nhớ chỉ có OS và một tiến trình người dùng. Tiến trình người dùng sẽ được toàn quyền trên phần bộ nhớ dành cho nó, khi nó kết thúc tiến trình khác sẽ được nạp đè lên và tiếp tục được toàn quyền trên vùng nhớ đó.

Trong rất nhiều trường hợp, hệ thống đơn chương không dùng được hoặc không hiệu quả, và ta phải sử dụng hệ thống đa chương. Khi này bộ nhớ phải được chia sẻ để nhiều tiến trình có thể sử dụng cùng lúc, việc quản lý điều hành bộ nhớ lúc này sẽ phải rất tinh tế mới có thể đạt được hiệu quả tốt. Hệ thống Quản lý Bộ Nhớ khi này sẽ phải xây dựng các mô hình tổ chức quản lý phức tạp và có các cơ chế /thuật toán tối ưu cho việc sử dụng.

2 Các mô hình tổ chức quản lý bộ nhớ

RAM về cơ bản là 1 dãy các ô nhớ, mỗi ô có kích thước 01 byte, và được đánh địa chỉ theo cơ chế đơn giản từ 0 trở đi. Tuy nhiên số chân truyền nhận dữ liệu của CPU thuở ban đầu mới là 8 bit còn mấy chục năm qua đã nâng lên nhiều hơn hẳn (CPU trên máy cá nhân hàng chục năm qua là 64bit). Vì vậy việc truy xuất RAM của CPU mỗi lần là nhiều hơn 1 byte nhớ, và RAM giờ được tổ chức như một dãy các từ nhớ (word), mỗi từ nhớ có một địa chỉ cũng theo cơ chế đánh offset vị trí từ 0 trở đi.

2.1 Mô hình cấp phát liên tục

*** Hệ đơn chương:** giải quyết như trên

*** Hệ thống đa chương với các phân vùng cố định:**

Một cách đơn giản là chia bộ nhớ thành những phân vùng cố định (có thể không bằng nhau). Các tiến trình có nhu cầu bộ nhớ được đặt vào một hàng đợi. Có thể mỗi phân vùng có một hàng đợi riêng, hoặc dùng một hàng đợi chung cho tất cả các phân vùng

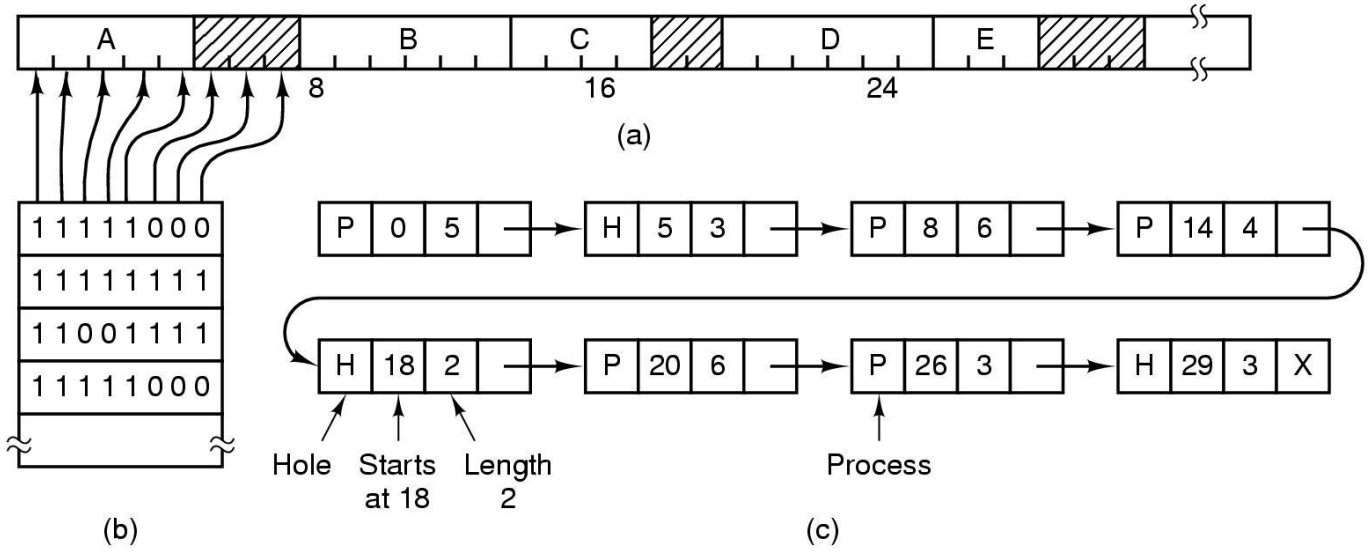
*** Hệ thống đa chương với các phân vùng động:**

Khi tiến trình được đưa vào hệ thống, một vùng nhớ trống vừa đúng với kích thước của nó sẽ được cấp phát cho nó, các vùng nhớ trống còn lại được dùng cho các tiến trình khác. Khi tiến trình kết thúc, vùng nhớ tương ứng của nó được giải phóng thành vùng nhớ trống.

Cơ chế quản lý chỗ trống:

- Bitmap: bộ nhớ được chia thành những đơn vị cấp phát, mỗi bit trên dãy bitmap quản lý một đơn vị, theo qui ước: bit 0 nghĩa là đơn vị đó còn trống, bit 1 nghĩa là đã được cấp phát.
- Danh sách liên kết: mỗi phần tử của danh sách liên kết quản lý một vùng trên bộ nhớ, mỗi phần tử có các thông tin: trạng thái vùng (H: trống, P: tiến trình /không trống), vị trí bắt đầu, kích thước, vị trí vùng kế tiếp.

Ví dụ:



Các thuật toán lựa chọn vùng trống để cấp phát:

- First-fit
- Last-fit
- Worst-fit
- Best-fit

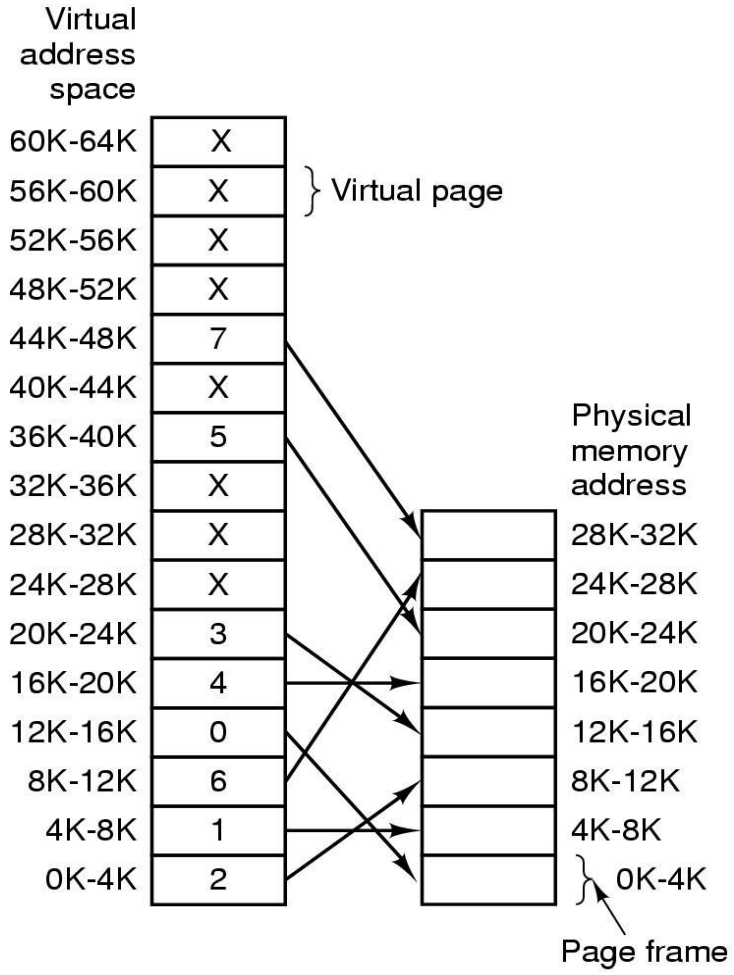
*** Swapping:**

Khi có nhiều tiến trình thì có thể không đủ bộ nhớ để chứa tất cả các tiến trình, vì vậy các tiến trình ở trạng thái chờ hơi lâu sẽ được tạm chuyển ra bộ nhớ phụ (khi nó được kích hoạt thì sẽ nạp lại vào bộ nhớ chính), kỹ thuật này gọi là swapping, Một kỹ thuật khác gọi là bộ nhớ ảo (virtual memory), cho phép chương trình chạy khi chỉ có một phần của chương trình được đưa vào bộ nhớ.

1.1 Mô hình cấp phát không liên tục

*** Phân trang (paging)**

Bộ nhớ vật lý được chia thành những khối (block) kích thước cố định và bằng nhau gọi là các khung trang (page frame), không gian địa chỉ được chia thành các khối cùng kích thước với khung trang, gọi là các trang (page). Khi cần nạp một tiến trình vào, các trang của tiến trình sẽ nạp vào các khung trang còn trống.



*** Phân đoạn (segmentation)**

Trong nhiều trương hợp nếu có nhiều không gian địa chỉ sẽ tốt hơn nếu chỉ có một không gian địa chỉ như ở cơ chế phân trang. Hệ thống phân đoạn có thể có nhiều không gian địa chỉ độc lập nhau, được gọi là các segments. Mỗi segment gồm một chuỗi các địa chỉ liên tục từ 0. Chiều dài của các segment có thể khác nhau. Tuy nhiên chiều dài của segment có thể thay đổi trong lúc chạy.

Ví dụ: chiều của segment dùng làm stack thay đổi mỗi khi đẩy dữ liệu vào stack, và giảm khi lấy dữ liệu ra khỏi stack.

Bởi vì mỗi segment có không gian địa chỉ khác nhau, các segment có thể to hoặc bé đi một cách độc lập nhau, không ảnh hưởng đến nhau.

Để xác định địa chỉ trong bộ nhớ được chia thành nhiều segment, địa chỉ gồm có hai phần, phần đầu là số segment, xác định segment được truy xuất, và phần còn lại của địa chỉ xác định vị trí bộ nhớ trong segment đó.