

Lý thuyết hệ điều hành

Giảng viên: TS. Hà Chí Trung

Bộ môn: Khoa học máy tính

Khoa: Công nghệ thông tin

Học viện Kỹ thuật quân sự

Email: hct2009@yahoo.com

Mobile: 01685.582.102

Chương 1. Tổng quan về hệ điều hành

1.1. Tổng quan về môn học

1.2. Khái niệm, chức năng hệ điều hành

1.3. Lịch sử phát triển của hệ điều hành

1.4. Cấu trúc của hệ thống máy tính

1.5. Một số khái niệm của hệ điều hành

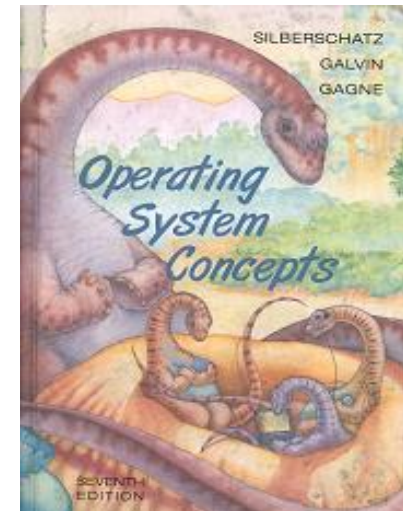
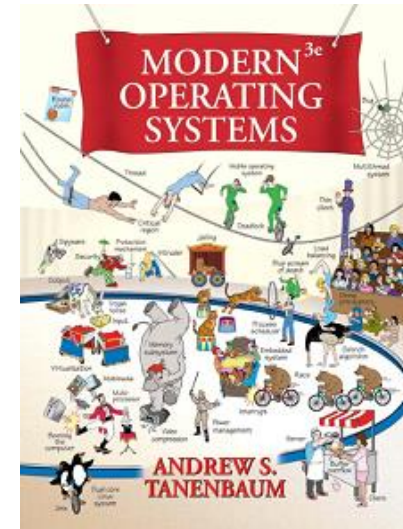
1.6. Cấu trúc Hệ điều hành

1.7. Một số hệ điều hành phổ biến

1.8. Bài tập cài đặt hệ điều hành, máy ảo

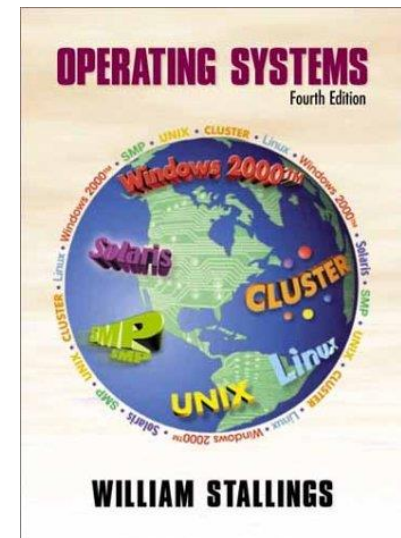
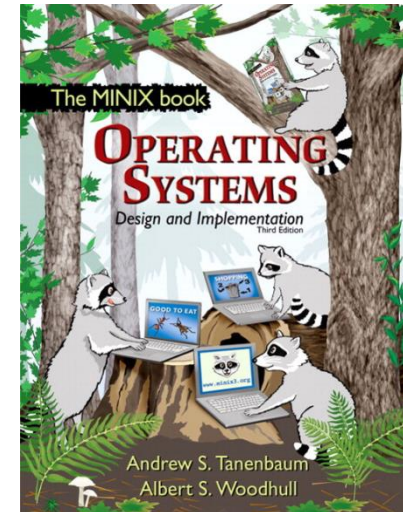
1.1. Tổng quan về môn học

- Tài liệu tham khảo:
 - **Modern Operating Systems, 3rd Edition**, Andrew S. Tanenbaum. Prentice-Hall, 2007.
 - **Operating System Concepts, 7th Edition**, Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, John Wiley & SONS, 2005.



1.1. Tổng quan về môn học

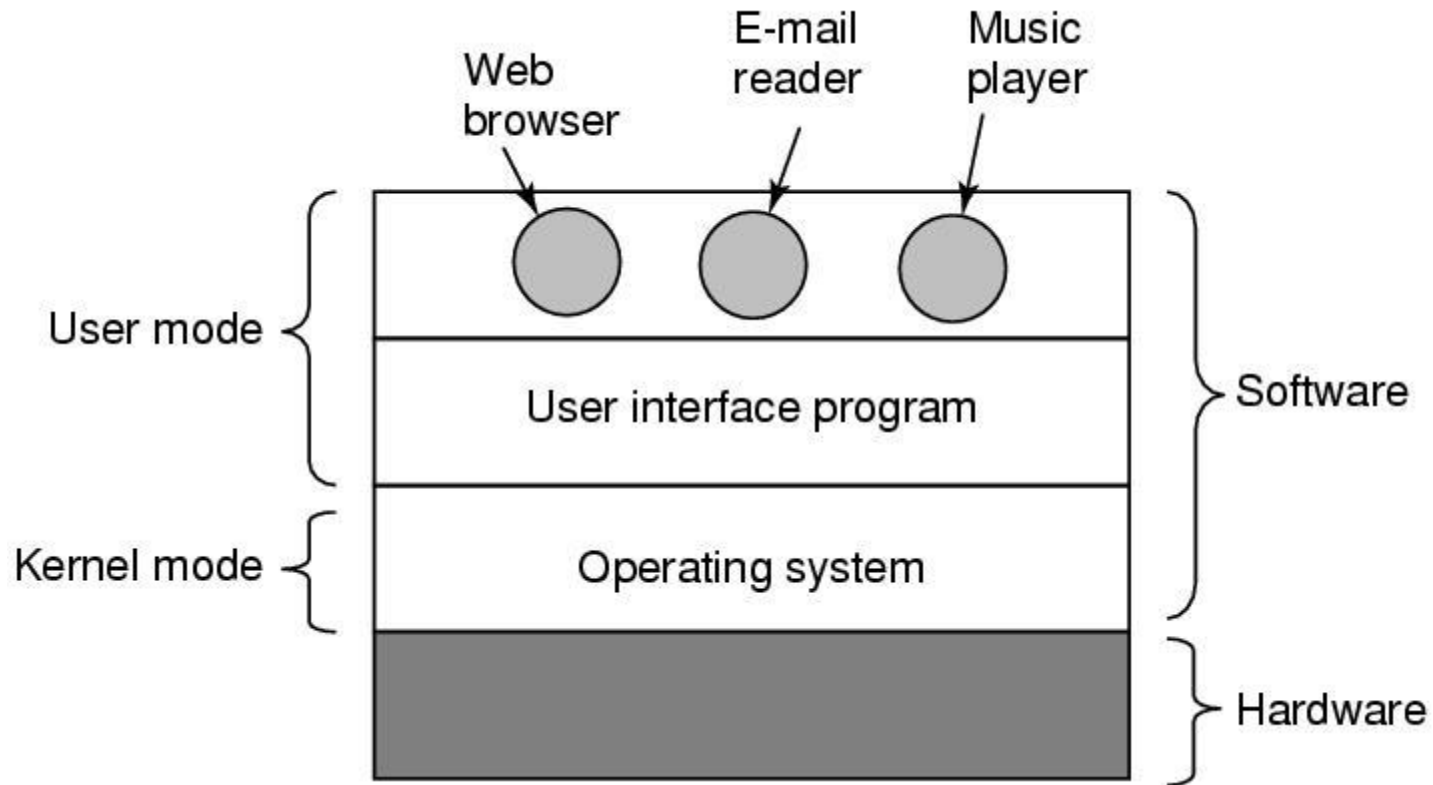
- Tài liệu tham khảo:
 - **Operating Systems Design and Implementation**, Third Edition. Andrew S. Tanenbaum, Albert S. Woodhull. Prentice Hallm, 2006.
 - **Operating Systems: Internals and Design Principles**, 4/E. William Stallings. 2000.
 - **Nguyên lý Hệ điều hành**, NXB KH&KT, 2005, Nguyễn Gia Định, Nguyễn Kim Tuấn.



1.2. Khái niệm, chức năng hệ điều hành

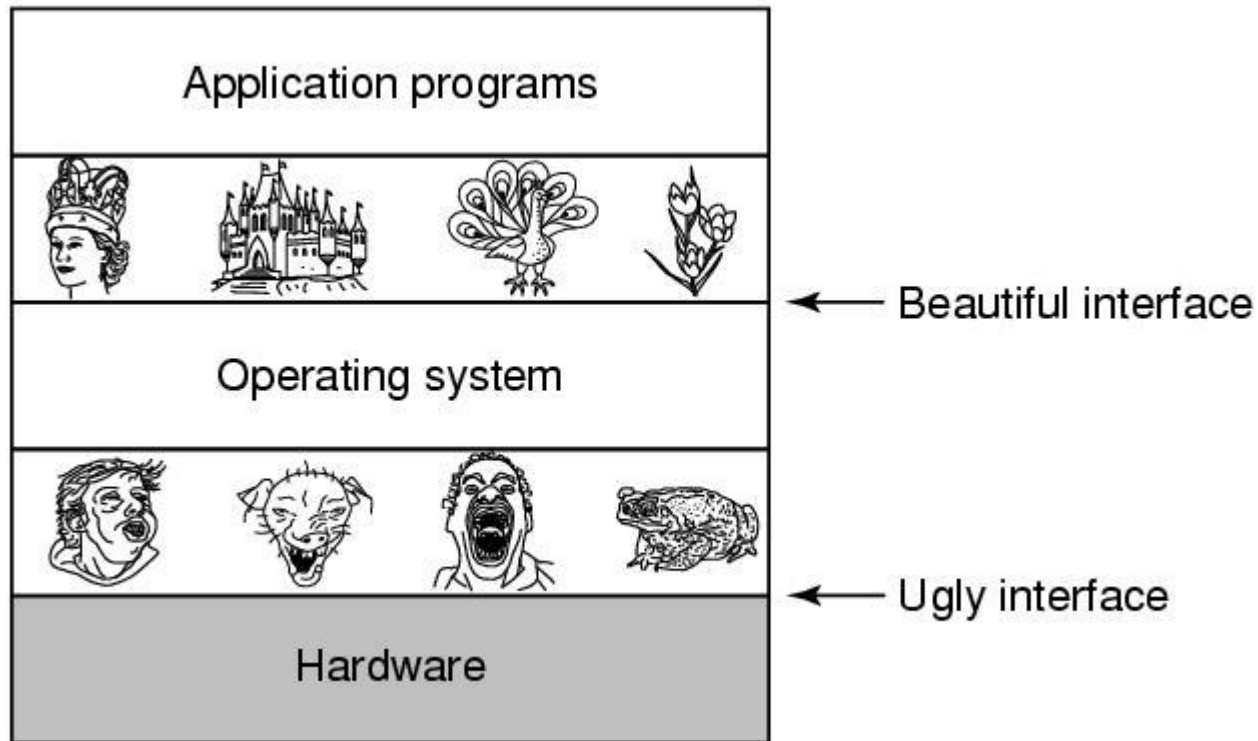
- Một máy tính hiện đại bao gồm:
 - Một hoặc một số bộ vi xử lý (processors)
 - Bộ nhớ (Main memory)
 - Các đĩa lưu trữ (Disks)
 - Máy in (Printers)
 - Các thiết bị vào ra khác
- Để quản lý tất cả các thành phần đó sao cho chúng tương tác, hoạt động với nhau, phục vụ cho mục đích của người dùng, cần có một phần mềm đặc biệt – hệ điều hành (operating system)

1.2. Khái niệm, chức năng hệ điều hành



- Vị trí của hệ điều hành trong hệ thống máy tính

1.2. Khái niệm, chức năng hệ điều hành



- Hệ điều hành – mở rộng của phần cứng vô tri

1.2. Khái niệm, chức năng hệ điều hành

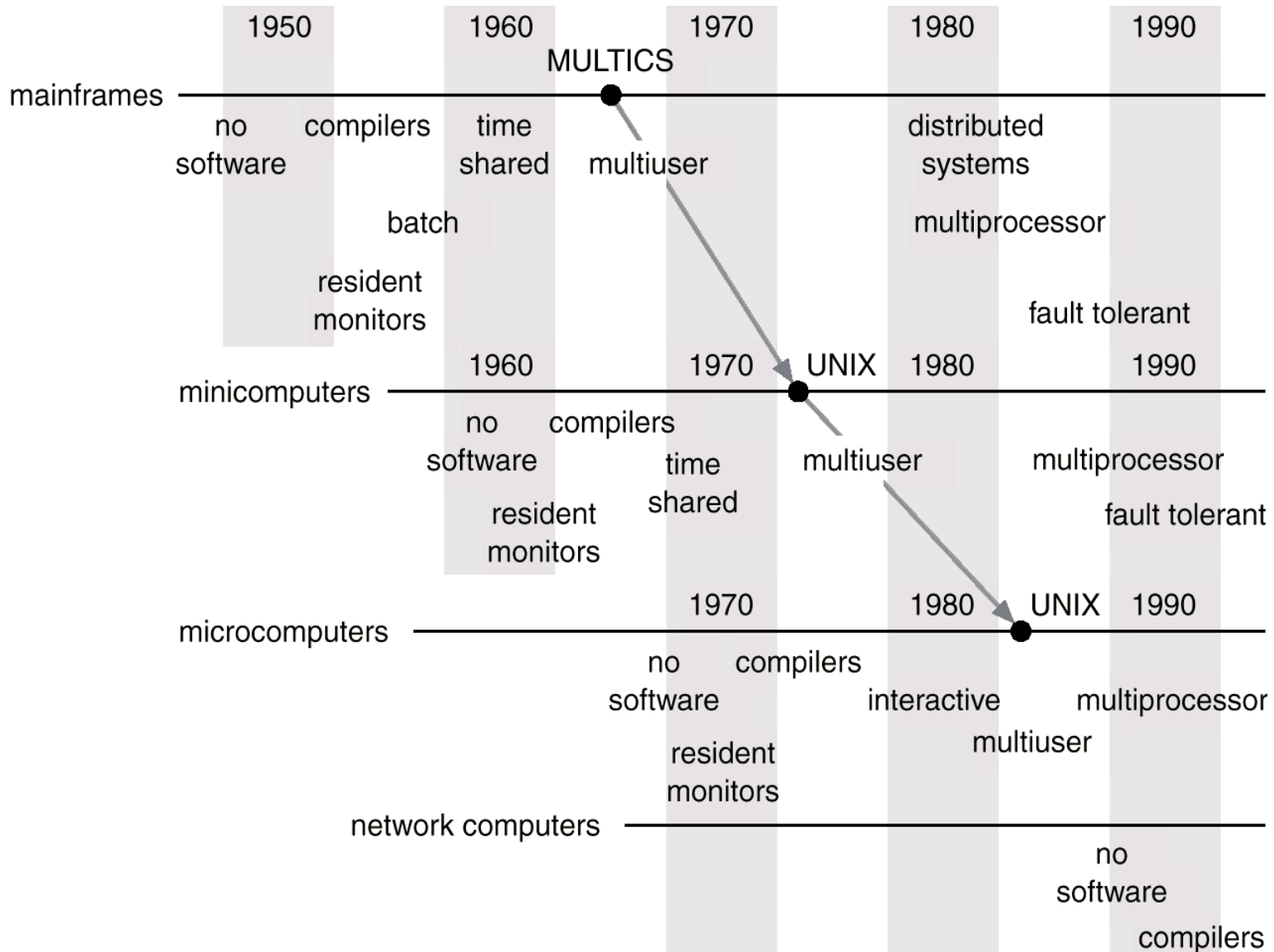
- **Khái niệm:**

- Một chương trình làm việc như người trung gian giữa người sử dụng và phần cứng máy tính.
- Một chương trình nào đó, viết trên C hay C++.
- Dùng để nạp và chạy các chương trình khác;
- Làm việc trên những gì mà phần cứng có.
- Phần mềm đảm bảo cơ sở cho các chương trình ứng dụng được thực thi:
 - Vẽ trên màn hình;
 - Tương tác qua bàn phím, chuột, ...;
 - Truy cập đến ổ đĩa, files;
 - Tương tác với các chương trình khác, máy tính khác...

1.2. Khái niệm, chức năng hệ điều hành

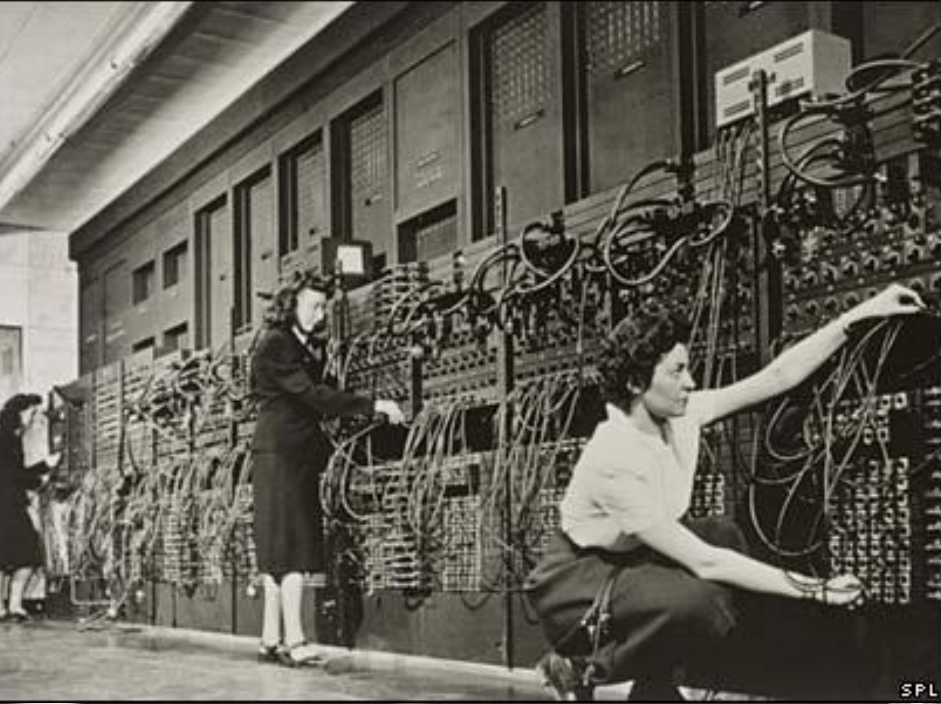
- **Chức năng:**
 - **Quản lý và phân bố** các tài nguyên hệ thống;
 - **Trình điều khiển:** quản lý và điều khiển việc thực thi các chương trình và giúp cho việc giải quyết các vấn đề của người dùng tốt hơn; Làm cho việc sử dụng hệ thống máy tính trở nên thuận lợi. Sử dụng phần cứng đạt hiệu quả;

1.3. Lịch sử phát triển hệ điều hành

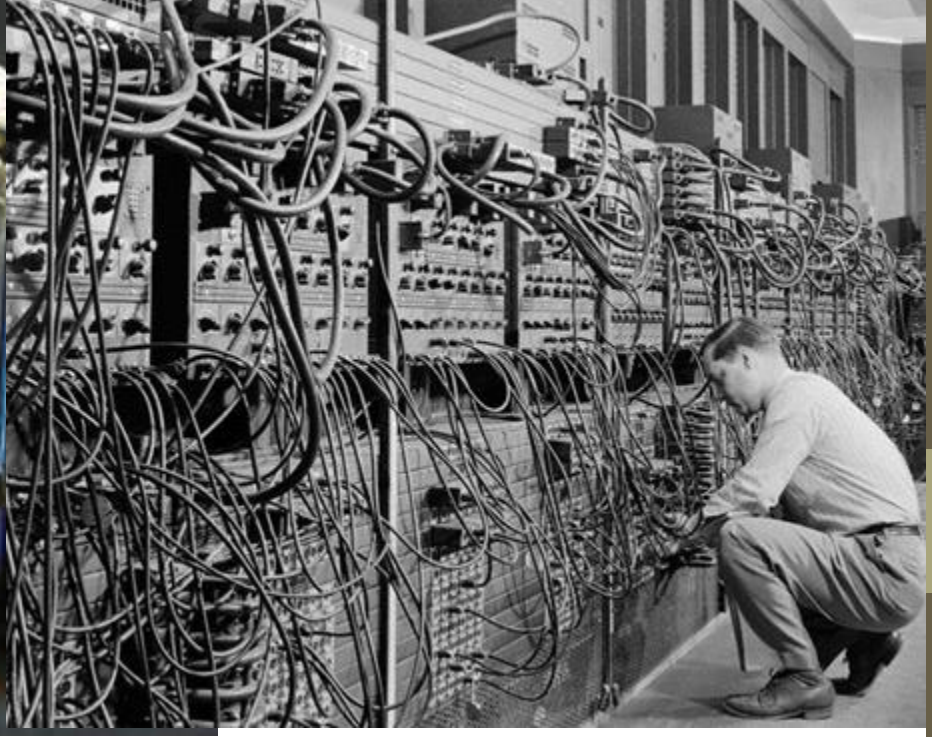
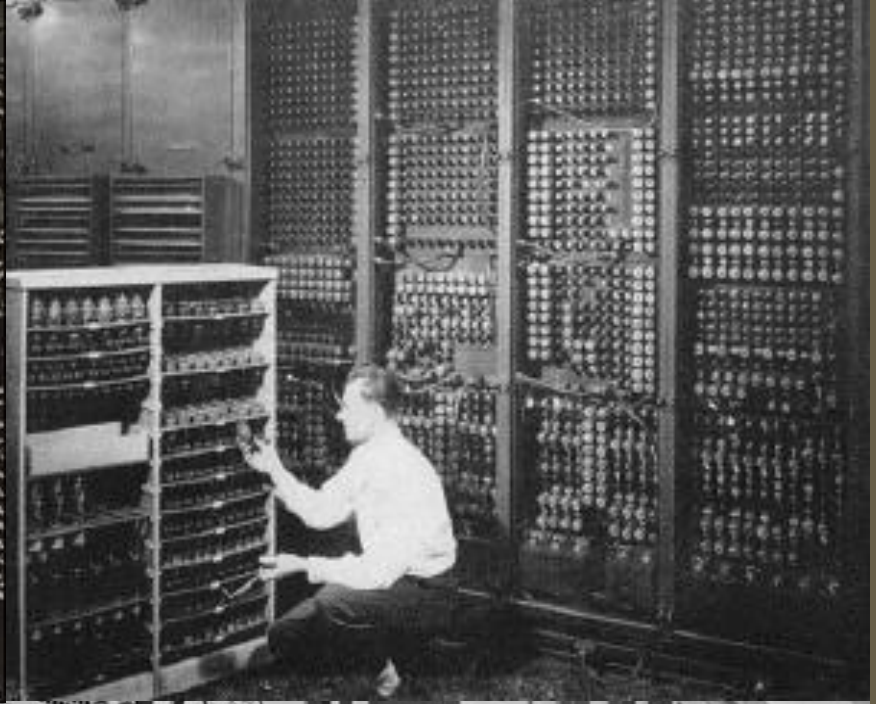


1.3. Lịch sử phát triển hệ điều hành

- Máy tính điện tử đầu tiên ENIAC (Electronic Numerical Integrator and Computer), tại Maryland, Hoa Kỳ.
 - Xây dựng trong những năm 1940s, nặng 30 tấn,
 - Cao 2.5m, sâu gần 1 m, dài hơn 30 m.
 - Bên trong chứa hơn 18,000 bóng chân không, làm nguội bởi hệ thống 80 quạt gió lớn;
 - Chưa có hệ điều hành;
 - Nhân viên vận hành (user operators).



SPL



1.3. Lịch sử phát triển hệ điều hành

- **Máy tính lớn (mainframes)** – được chia làm 3 loại:
 - Xử lý bó đơn (Simple batch)
 - Đa chương (multiprogrammed)
 - Đa nhiệm (time-sharing, multitasking)
- **Máy tính cá nhân (Desktop system)**
- **Các hệ thống song song**
- **Các hệ thống phân tán**
- **Các hệ thống xử lý thời gian thực**
- V.v.

1.3. Lịch sử phát triển hệ điều hành

1.3.1. Xử lý bó đơn (Simple batch systems)

1.3.2. Multiprogrammed systems

1.3.3. Hệ chia sẻ thời gian (Time-sharing systems)

1.3.4. Máy để bàn (Desktop systems)

1.3.5. Các hệ thống song song

1.3.6. Hệ phân tán (Distributed system)

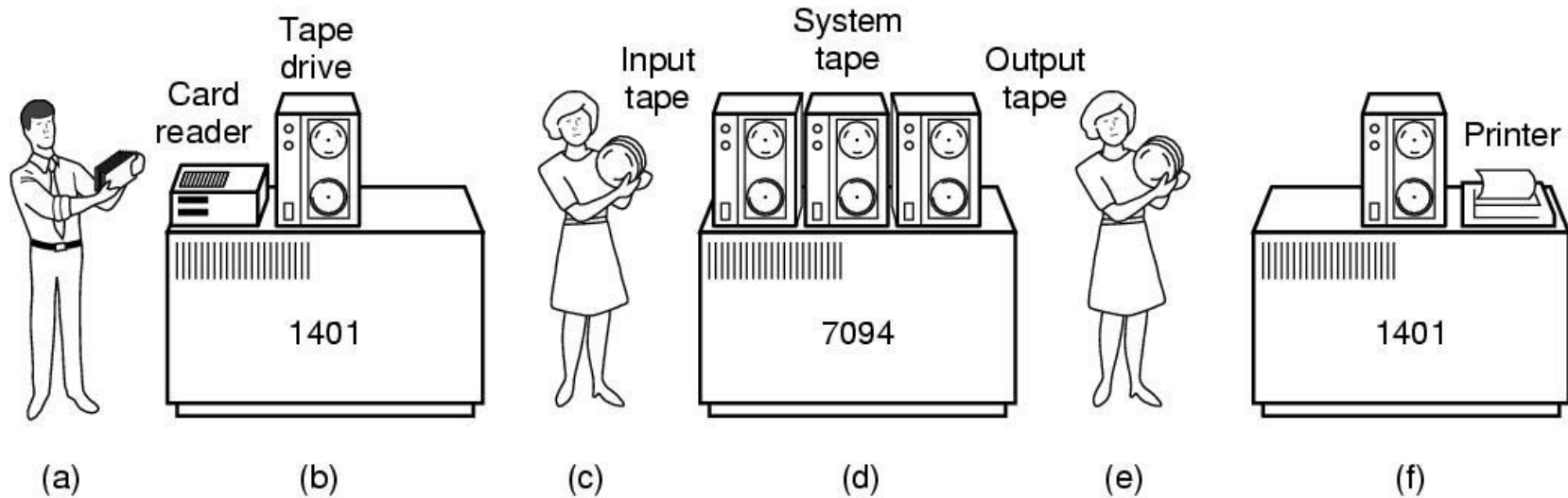
1.3.7. Các hệ thống thời gian thực

1.3.1. Xử lý bó đơn (Simple batch)

Mainframe xử lý bó đơn (Simple batch systems) :

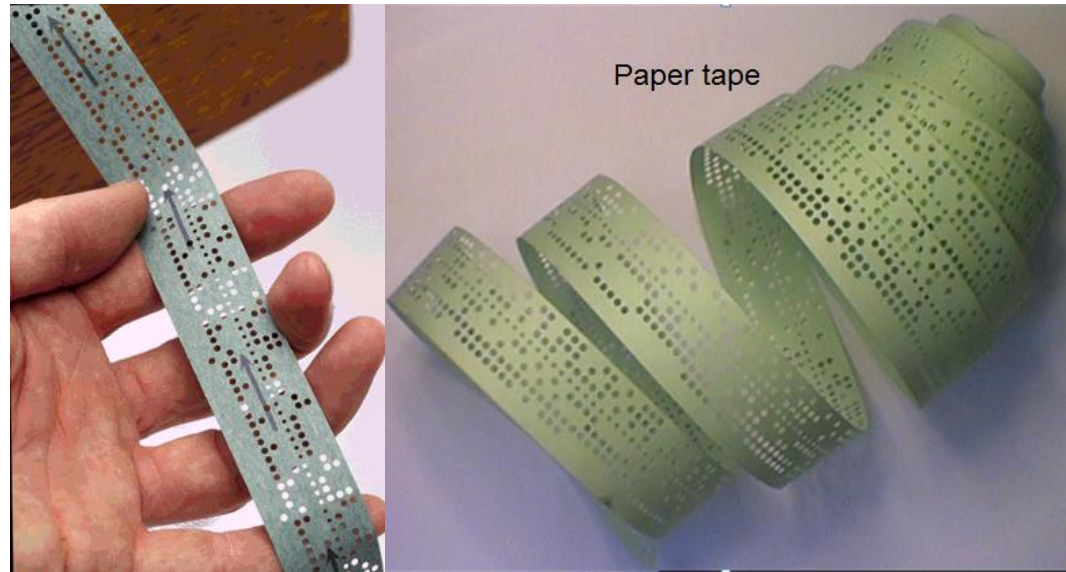
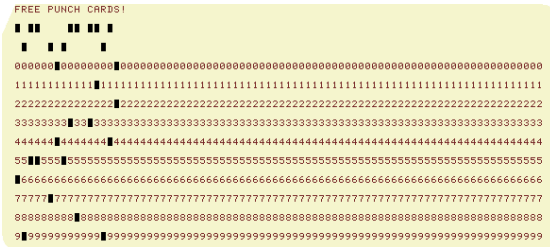
- Chương trình và kết quả được nhập/xuất bằng bìa đục lỗ.
- Cần có người vận hành (user operator) để đóng ngắt các mạch điện;
- Để giảm thời gian khởi tạo (setup time) người ta sử dụng kỹ thuật viên ghép nhóm công việc (batching) tương tự nhau;
- Hệ thống tự động nạp và thực hiện các chương trình theo thứ tự;
- **Hạn chế:** không cung cấp khả năng tương tác với users trong quá trình hoạt động.

1.3.1. Xử lý bó đơn (Simple batch)

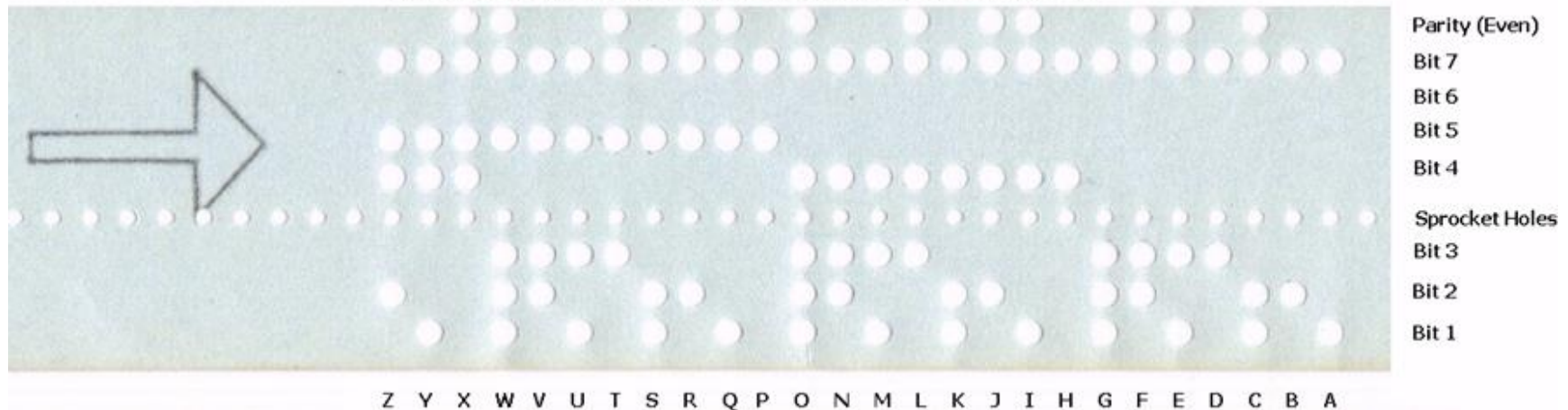


(a) – lập trình viên mang thẻ đến máy đọc thẻ; (b) máy đọc thẻ gom các thẻ và xuất ra băng; (c) kỹ thuật viên chuyển băng sang máy tính; (d) máy tính tính toán; (e) kỹ thuật viên chuyển kết quả sang máy in; (f) máy in in kết quả từ băng.

1.3.1. Xử lý bó đơn (Simple batch)



Punched Paper Tape 25.4 mm wide. Ascii 7-bit character code. Even Parity.



1.3.2. Multiprogrammed systems

Mainframe thực hiện nhiều chương trình:

- Nhiều công việc được nạp đồng thời vào bộ nhớ chính
- Thời gian xử lý của CPU được phân chia giữa các cv đó
- Các yêu cầu nhất thiết của hệ thống:
 - Lập lịch cho các công việc (**job scheduling**);
 - Lập lịch cho CPU (**CPU scheduling**) – phân chia CPU cho các tiến trình.
- **Hạn chế:** không cung cấp khả năng tương tác với users trong quá trình hoạt động
 - VD: IBM OS/360, MULTICS, DOS/VSE

1.3.3. Time-sharing systems

Mainframe chia sẻ thời gian

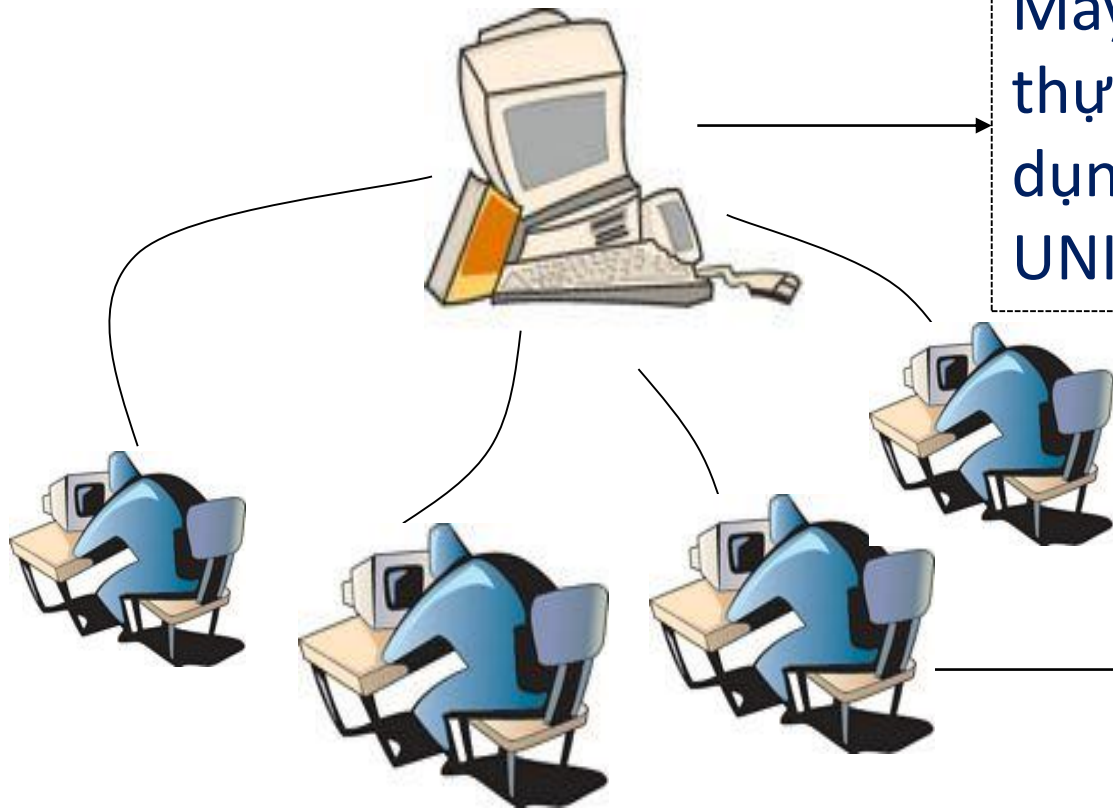
- Time-sharing là một sự phát triển tiếp theo của hệ thống đa chương. Nó cho phép người sử dụng tương tác với hệ thống trong quá trình hoạt động;
- CPU luân phiên thực thi giữa các công việc;
- Một công việc chỉ được chiếm CPU khi nó nằm trong bộ nhớ chính;
- Khi cần thiết, một công việc nào đó có thể được chuyển từ bộ nhớ chính ra thiết bị lưu trữ thứ cấp (swapping), nhường bộ nhớ chính cho công việc khác.

1.3.3. Time-sharing systems

Mainframe chia sẻ thời gian:

- Yêu cầu nhất thiết với OS trong hệ thống time-sharing
 - Định thời gian cho công việc (job scheduling)
 - Định thời gian cho CPU (CPU scheduling)
 - Quản lý bộ nhớ (memory management)
 - Quản lý các tiến trình (process management)
 - Giao tiếp giữa các tiến trình (process communication)
 - Đồng bộ các tiến trình (synchronization)
 - Ngăn ngừa deadlock
 - Quản lý hệ thống file, hệ thống lưu trữ
 - Cấp phát hợp lý các tài nguyên
 - Bảo vệ (protection) các tài nguyên của hệ thống.
- VD: IBM OS/VS1, UNIX, IBM/400, Windows 9x/NT/2000

1.3.3. Time-sharing systems



Máy chủ, có một CPU thực hiện quá trình sử dụng hệ điều hành như UNIX.

Thiết bị đầu cuối được kết nối với máy chủ và được sử dụng cho đầu vào và đầu ra dữ liệu. Xử lý không được thực hiện. Không có CPU.

1.3.4. Máy để bàn (Desktop systems)

Máy để bàn (desktop system, personal computer)

- Có nhiều thiết bị I/O: bàn phím, chuột, màn hình, máy in,...
- Người dùng là từng cá nhân đơn lẻ.
- Mục tiêu chính của OS dành cho máy cá nhân:
 - Thuận tiện cho user và khả năng tương tác cao.
 - Không cần tối ưu hiệu suất sử dụng CPU và thiết bị ngoại vi.
- VD: MS-DOS, Windows 3.1, Windows 95, 98, Ubuntu, XP, Windows 7,...

1.3.4. Máy để bàn (Desktop systems)

Đặc điểm:

- Đa chương;
 - Khả năng xử lý đồng thời, trực tuyến với các thiết bị ngoại vi;
 - Time sharing;
 - Khả năng lưu trữ trực tuyến,
 - Sử dụng các chương trình, dữ liệu, thư viện chương trình;
 - Sử dụng bộ nhớ ảo,
 - Cấu hình Multiprocessor.
-
- VD: System 360 and S/370;
Commodore PET, 1977



1.3.5. Các hệ thống song song

- **Hệ song song:**
 - Nhiều CPU;
 - Chia sẻ chung đường bus và đồng hồ thời gian (clock)
- **Ưu điểm:**
 - Tốc độ xử lý nhanh do có thể tận dụng nhiều processor cùng thực hiện một chương trình (phân ra thành các threads);
 - Multiprocessor system ít tốn kém hơn multiple single-processor system: vì có thể dùng chung tài nguyên (đĩa, băng từ...);
 - Độ tin cậy: khi một processor hỏng thì công việc của nó được chia sẻ giữa các processor còn lại.

1.3.5. Các hệ thống song song

- **Phân loại hệ thống song song**
 - **Đa xử lý đối xứng (symmetric multiprocessor - SMP)**
 - Mỗi bộ xử lý vận hành một bản copy của hệ điều hành
 - Các bản copy của hệ điều hành có thể giao tiếp được với nhau khi cần (VD: HĐH Windows 2000).
 - **Đa xử lý bất đối xứng (asymmetric multiprocessor)**
 - Có một processor chủ (master processor);
 - Mỗi processor phụ (slave processors) thực thi một công việc khác nhau;
 - Master processor định thời và phân công việc cho các processors phụ.

1.3.6. Hệ phân tán (Distributed systems)

Hệ thống phân tán (Distributed systems; loosely-coupled systems)

- Mỗi processor có bộ nhớ riêng, các bộ xử lý giao tiếp với nhau qua các kênh nối tốc độ cao.
- Người dùng chỉ thấy một hệ thống đơn nhất
- Ưu điểm
 - Chia sẻ tài nguyên (resource sharing)
 - Chia sẻ sức mạnh tính toán (computational sharing)
 - Độ tin cậy cao (high reliability)
 - Độ sẵn sàng cao (high availability): các dịch vụ của hệ thống được cung cấp liên tục cho dù một bộ xử lý nào đó có bị hỏng

1.3.6. Hệ phân tán (Distributed system)

- Các mô hình hệ thống phân tán:
 - **Client-server**
 - Server: cung cấp dịch vụ;
 - Client: có thể sử dụng dịch vụ của server
 - **Peer-to-peer (P2P)**
 - Các máy tính trong hệ thống đều ngang hàng nhau
 - Không có cơ sở dữ liệu tập trung
 - Vd: Gnutella – là một chương trình xuất hiện vào tháng 3 năm 2004 có kích thước chỉ 104 KB cho phép chia sẻ tất cả các loại files với người sử dụng khác có Gnutella.

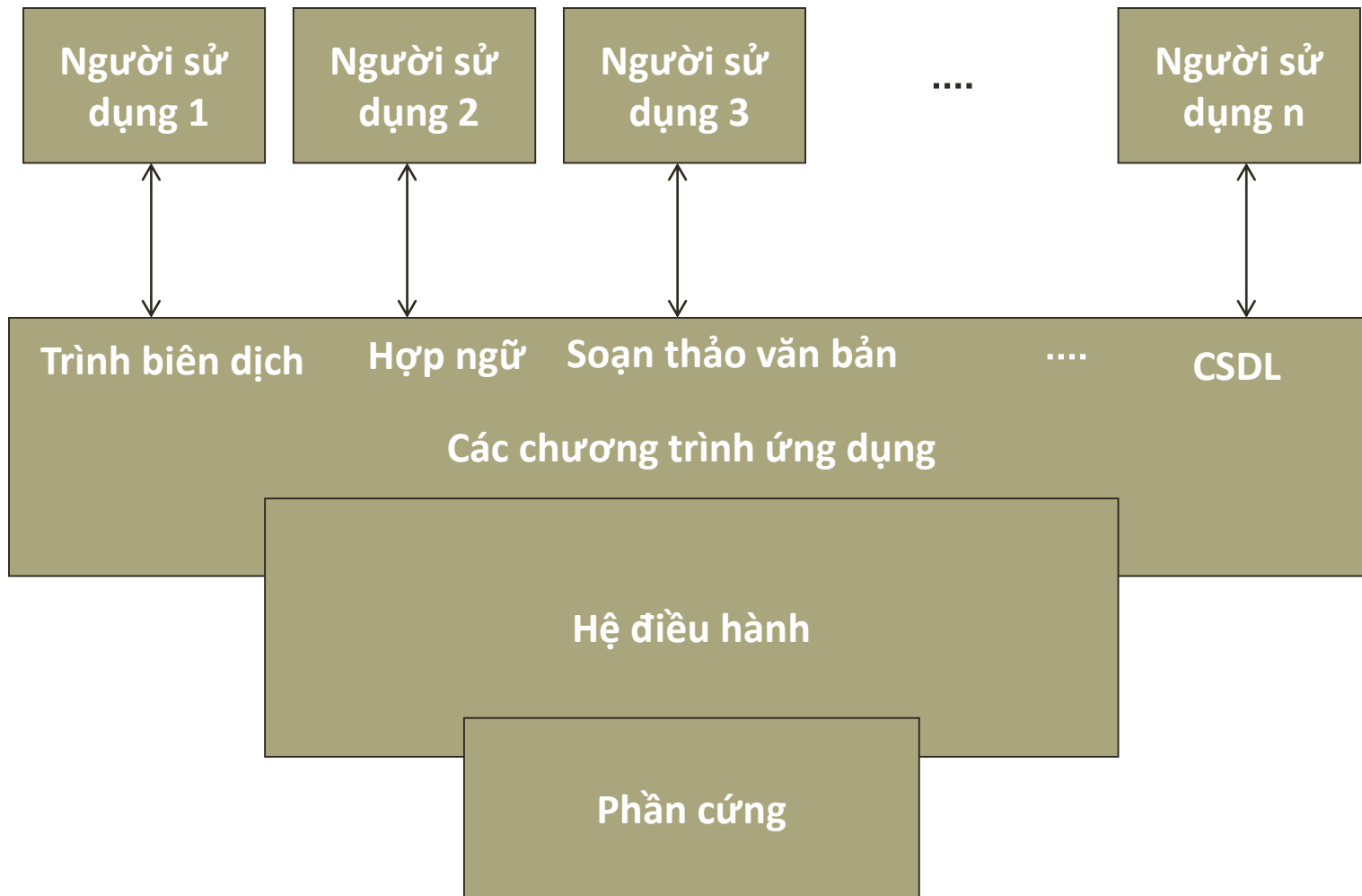
1.3.7. Các hệ thống thời gian thực

- Sử dụng trong các thiết bị chuyên dụng như điều khiển các thử nghiệm khoa học, điều khiển trong y khoa, dây chuyền công nghiệp, thiết bị gia dụng...
- Có 2 loại ràng buộc : **hard real-time** và **soft real-time**.
- **Hard real-time:**
 - Yêu cầu công việc phải kết thúc đúng giờ quy định
 - Hạn chế (hoặc không có) bộ nhớ phụ, tất cả dữ liệu nằm trong bộ nhớ chính (RAM hoặc ROM)
 - Yêu cầu về thời gian đáp ứng/xử lý rất nghiêm ngặt, thường sử dụng trong điều khiển công nghiệp, robotics,...
- **Soft real-time:**
 - Các cv cần realtime sẽ được thực hiện với độ ưu tiên cao nhất.
 - Thường được dùng trong lĩnh vực multimedia, thực tại ảo (virtual reality) với yêu cầu mềm dẻo hơn về thời gian đáp ứng

1.4.1. Cấu trúc của hệ thống máy tính

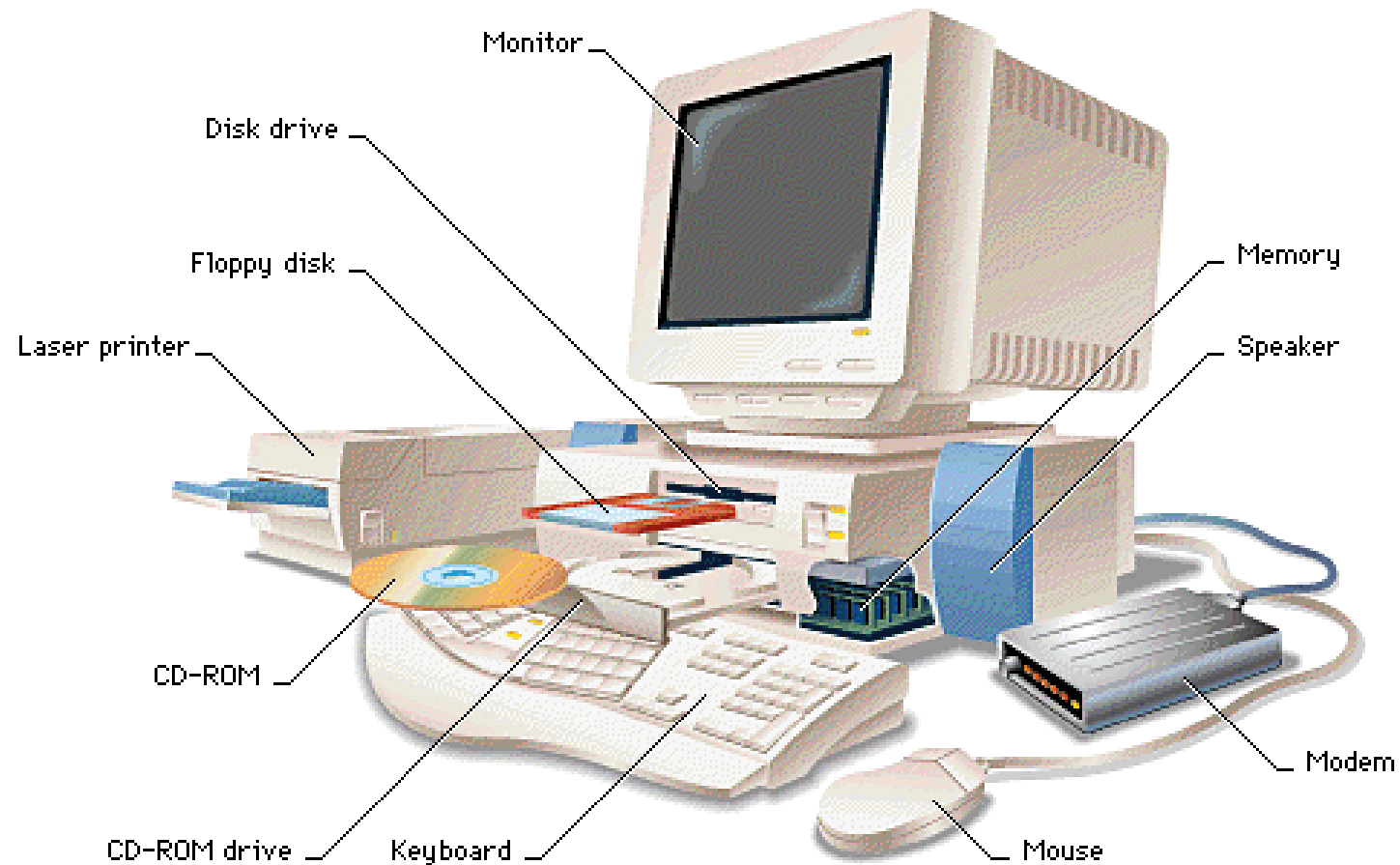
- **Phần cứng (hardware):** CPU; Bộ nhớ: RAM, ROM, đĩa từ, băng từ...; Thiết bị I/O: Màn hình, bàn phím, card I/O...
- **Phần mềm (software):**
 - Phần mềm hệ thống
 - Hệ quản trị CSDL: Oracle, SQL Server
 - Tiện ích: Norton Disk Doctor, Window Command, ..
 - Phần mềm ứng dụng: MS Office, Corel Draw, Netscape Navigator,...
 - Ngôn ngữ lập trình: C, C++, C#...
- **Hệ điều hành:** {MS-DOS, Windows 9x/ NT/ ME/ 2000/ XP...}; {Linux, Solaris, HP-UX, AIX, BSD, MacOS,...} {Novell Netware}
- **Người sử dụng:**
 - Có nhiều loại người dùng khác nhau, thực hiện những yêu cầu khác nhau, tính chất công việc khác nhau, do đó sẽ có nhiều ứng dụng khác nhau

1.4. Cấu trúc của hệ thống máy tính



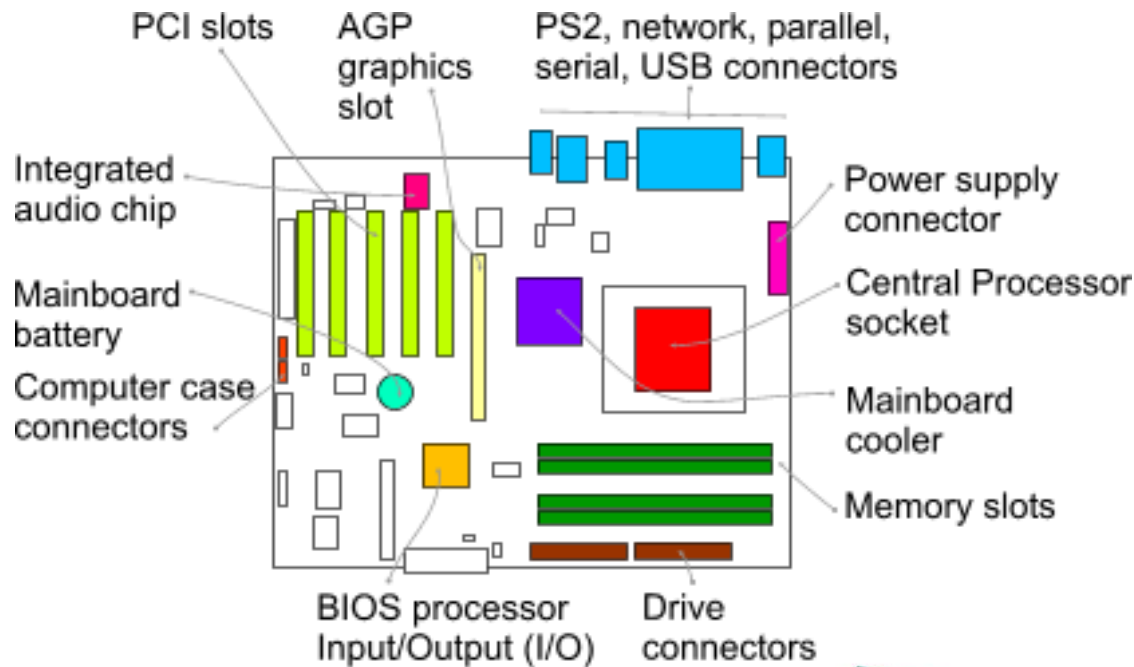
1.4. Cấu trúc của hệ thống máy tính

- Kiến trúc máy tính:



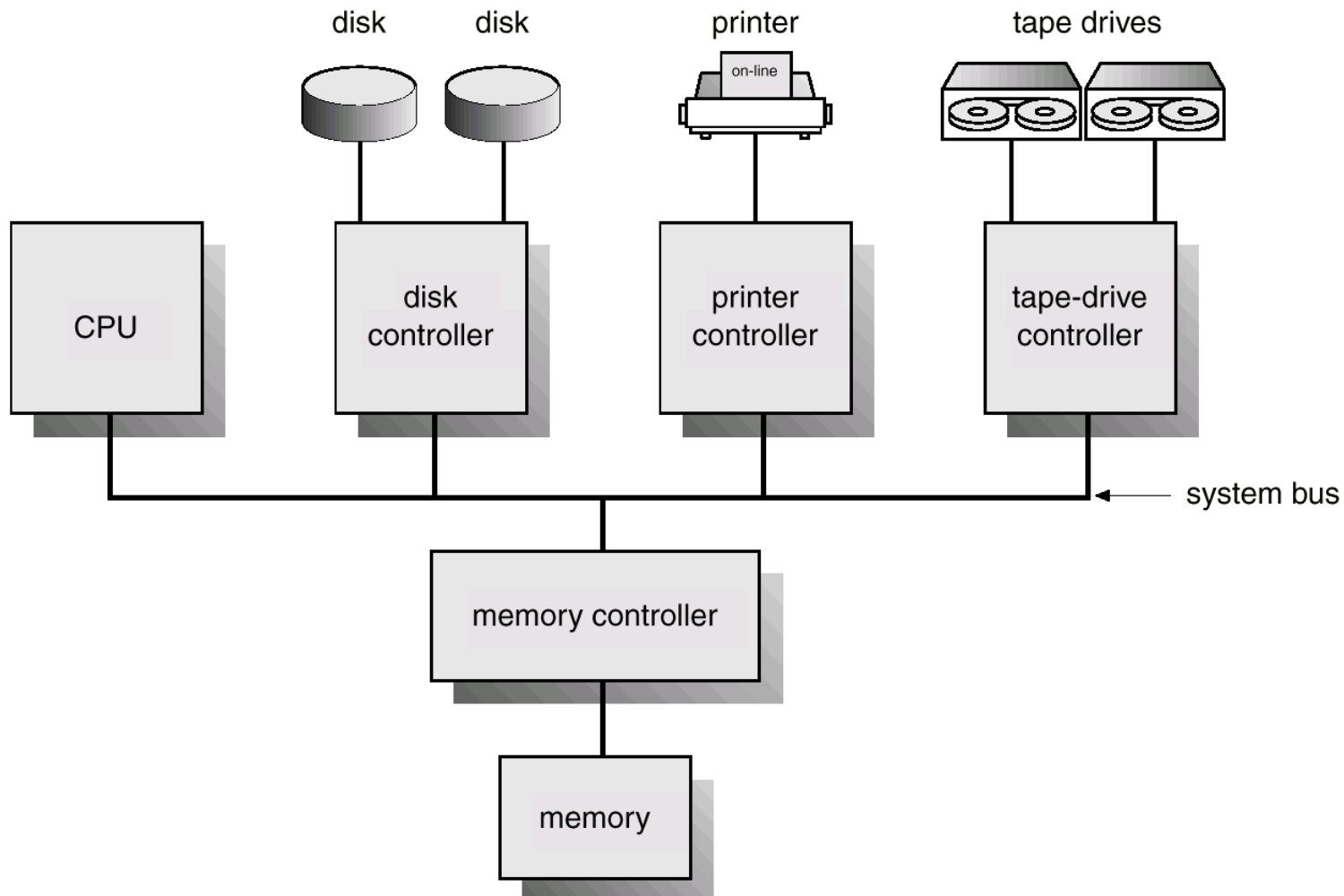
1.4. Cấu trúc của hệ thống máy tính

- Kiến trúc máy tính:



1.4. Cấu trúc của hệ thống máy tính

- Kiến trúc máy tính:



1.5. Một số khái niệm của hệ điều hành

1.5.1. Quản lý tiến trình (process management)

1.5.2. Quản lý bộ nhớ chính

1.5.3. Quản lý hệ thống nhập/xuất (I/O system management)

1.5.4. Quản lý file (file management)

1.5.5. Hệ thống bảo vệ

1.5.6. Trình thông dịch lệnh

1.5.7. Các loại lời gọi hệ thống

1.5.1. Quản lý tiến trình

- Tiến trình vs. chương trình: tiến trình là chương trình đã được nạp vào trong bộ nhớ chính
- Một tiến trình cần các tài nguyên của hệ thống như CPU, bộ nhớ, file, thiết bị I/O,... để hoàn thành công việc. Khi kết thúc công việc tiến trình sẽ giải phóng các tài nguyên này

1.5.1. Quản lý tiến trình

- Các nhiệm vụ của thành phần quản lý tiến trình
 - Tạo và hủy tiến trình
 - Lập lịch thực thi các tiến trình
 - Thay đổi độ ưu tiên của tiến trình
 - Tạm ngưng/tiếp tục thực thi (suspend/resume) tiến trình
 - Cung cấp các cơ chế:
 - đồng bộ hoạt động các tiến trình (synchronization)
 - giao tiếp giữa các tiến trình (interprocess communication)
 - khổng chế deadlock

1.5.1. Quản lý tiến trình

- Hệ điều hành khởi tạo tiến trình theo thứ tự sau:
 - Cấp phát một định danh duy nhất (process number hay process identifier, pid) cho tiến trình.
 - Cấp phát không gian nhớ để nạp tiến trình.
 - Xác định mức độ ưu tiên của tiến trình và chèn thêm tiến trình vào hàng đợi Ready Jobs
 - Khởi tạo khối dữ liệu điều khiển tiến trình process control block (PCB) cho tiến trình. PCB là nơi hệ điều hành lưu các thông tin về tiến trình.
 - Một tiến trình có thể sinh ra các tiến trình con, do đó chúng ta sẽ có cấu trúc phân cấp hình cây để quản lý tiến trình.

1.5.1. Quản lý tiến trình

- Hủy bỏ tiến trình: Giải phóng các tài nguyên đã cấp phát cho tiến trình. Trong một số hệ thống khi ta hủy bỏ tiến trình cha thì các tiến trình con sẽ tự động bị hủy bỏ.
- Thay đổi độ ưu tiên của tiến trình: thông thường là tăng độ ưu tiên cho các tiến trình đã đợi lâu trong hàng đợi.

1.5.2. Quản lý bộ nhớ chính

- CPU chỉ có thể truy cập trực tiếp bộ nhớ chính do đó để thực hiện được thì chương trình phải được nạp vào bộ nhớ chính và HĐH sẽ quản lý chương trình thông qua địa chỉ vùng nhớ mà nó đã được nạp vào. Cùng một thời điểm trong các hệ thống đa nhiệm có thể có nhiều tiến trình trong bộ nhớ chính.
- **Các nhiệm vụ:**
 - Phân phối vùng nhớ cho các tiến trình khi có yêu cầu.
 - Theo dõi, quản lý các vùng nhớ: vùng nhớ nào còn trống và vùng nhớ nào đã cấp phát
 - Quyết định sẽ nạp chương trình nào khi có vùng nhớ trống.
 - Thu hồi các vùng nhớ đã được cấp phát

1.5.3. Quản lý hệ thống nhập/xuất

- Nhiệm vụ chính của HĐH là điều khiển các thiết bị nhập xuất thông qua các ngắt nhập xuất.
- Các ngắt nhập xuất được phần cứng khai báo.
- Ngắt thông báo cho CPU biết trạng thái của một kênh hay một thiết bị.
- HĐH che dấu sự phức tạp và các đặc trưng riêng biệt của từng thiết bị I/O tạo ra sự dễ dàng sử dụng nhất với người sử dụng.

1.5.4. Quản lý tệp tin (file management)

- Tệp tin là một tập hợp dữ liệu có tên riêng. Tệp tin được lưu trữ trong các bộ nhớ phụ như đĩa, băng từ.
- Các thao tác đối với tệp tin:
 - Mở tệp tin: chuẩn bị một tệp tin để sẵn sàng thao tác
 - Đóng tệp tin: Ngăn cản mọi thao tác liên quan đến tệp tin
 - Tạo/Hủy tệp tin: tạo ra một tệp tin mới, xóa bỏ một tệp tin
 - Xem nội dung tệp tin; Đổi tên tệp tin; Sao chép tệp tin.
- Xử lý dữ liệu trong tệp tin: Đọc/ghi; Cập nhật; Chèn; Xóa

1.5.4. Quản lý tập tin (file management)

- Các chức năng của hệ thống quản lý tập tin:
 - Cung cấp các phương pháp truy cập dữ liệu trong tập tin.
 - Quản lý tập tin: cung cấp các cơ chế lưu trữ, truy cập, dùng chung và bảo mật các tập tin.
 - Quản lý và cấp phát vùng nhớ cho các tập tin. Quản lý vùng trống (free space management); Cấp phát không gian lưu trữ (storage allocation); Lập lịch truy cập đĩa (disk scheduling))
 - Thực hiện cơ chế sao lưu, phục hồi tập tin.

1.5.5. Hệ thống bảo vệ

- Dùng chung và bảo vệ thông tin là hai mục đích trái ngược nhau. Khi mạng máy tính và yêu cầu chia sẻ thông tin càng phát triển thì nhiệm vụ của hệ thống bảo vệ càng phức tạp.
- **Bảo vệ mức vật lý:** ngăn cấm không cho người ngoài xâm phạm và làm hỏng học hệ thống máy tính.

1.5.5. Hệ thống bảo vệ

- **Bảo về mức hệ thống:** Đối với các hệ thống cho phép nhiều user hay nhiều tiến trình cùng truy cập hay cùng thực hiện, chúng ta cần:
 - Kiểm soát tiến trình người sử dụng đăng nhập/xuất và sử dụng hệ thống
 - Kiểm soát việc các tiến trình sử dụng tài nguyên trong hệ thống
 - Bảo đảm chỉ những người dùng/tiến trình đủ quyền hạn mới được phép sử dụng các tài nguyên tương ứng.
- **Các nhiệm vụ:**
 - Cung cấp cơ chế kiểm soát đăng nhập/xuất (login, log-out)
 - Phân định được sự truy cập tài nguyên hợp pháp và bất hợp pháp (authorized/unauthorized)

1.5.6. Trình thông dịch lệnh

- Là giao diện chủ yếu giữa người dùng và OS
 - Ví dụ: dòng nhập lệnh, dùng mouse, menu để thực hiện chương trình
- Khi user đã đăng nhập, bộ diễn giải dòng lệnh chạy (command interpreter hoặc shell) chạy, và chờ nhận lệnh từ người dùng, thực thi lệnh và gửi kết quả trả về.
- Trình thông dịch phải liên hệ chặt chẽ với các thành phần khác của hệ điều hành để thực thi các yêu cầu của người dùng
- VD: MSDOS: Command.com; Unix có các tiện ích Shell khác nhau, như Bourne Shell (chứa trong tập tin .sh), C Shell (.csh); Korn Shell (.ksh).

1.5.7. Các dạng lời gọi hệ thống

- **Lời gọi hệ thống điều khiển tiến trình:**
 - Kết thúc, huỷ bỏ tiến trình (end, abort); Nạp và thực hiện tiến trình (load, execute); Tạo và dừng tiến trình (create, terminate)
 - Đọc và đặt thuộc tính của tiến trình (get/set attributes);
 - Đợi tiến trình (wait);
 - Cấp và thu hồi bộ nhớ (allocate/ deallocate)
- **Quản lý tệp:**
 - Tạo và xoá tệp; Mở, đóng tệp;
 - Đọc, ghi, dịch chuyển dữ liệu tệp;
 - Đọc/ đặt thuộc tính tệp

1.5.7. Các loại lời gọi hệ thống

- **Quản lý thiết bị:**

- Yêu cầu và trả thiết bị (request/release); Đọc/ ghi dữ liệu
- Đọc/ đặt thuộc tính thiết bị; Gắn và tách thiết bị (attach/detach)

- **Bảo trì thông tin:**

- Lấy/ đặt thông tin thời gian hệ thống;
- Lấy/ đặt dữ liệu hệ thống

- **Truyền thông:**

- Tạo và giải phóng kết nối;
- Gửi/nhận thông tin trạng thái của kênh truyền thông

Hoạt động của hệ thống máy tính

- Khởi động: Để khởi động máy tính, một chương trình khởi động trong ROM được sử dụng. Chương trình này sẽ tìm kiếm vị trí HĐH và nạp HĐH vào bộ nhớ và trao quyền điều khiển cho HĐH.
- Xử lý các lệnh tuần tự trong các tiến trình
- Xử lý các sự kiện. Sự xuất hiện của các sự kiện thường được thông báo bởi các ngắt (Interrupts).
 - Ngắt cứng – do phần cứng phát ra một ngắt tại bất cứ thời điểm nào bằng cách gửi một tín hiệu tới CPU thông qua đường bus hệ thống. VD: Các thiết bị I/O phát tín hiệu kết thúc công việc nhập/xuất dữ liệu.

Hoạt động của hệ thống máy tính

- Ngắt thời gian - cho phép CPU thực thi một tác vụ nào đó theo định kỳ.
- Ngắt mềm –do chương trình phát ra bằng cách thực hiện một lời gọi hệ thống (system call) vd.: Read, Write files.
- Bên trong HĐH, mỗi một ngắt được điều khiển bởi một tiến trình phục vụ ngắt (Interrupt service routine).
- Khi CPU bị ngắt, nó sẽ dừng công việc hiện tại mà truyền điều khiển tới tiến trình phục vụ ngắt. Một bảng vector ngắt được dùng để xác định tiến trình phục vụ ngắt cho các ngắt tương ứng.

1.6. Cấu trúc của Hệ điều hành

1.6.1. Cấu trúc đơn giản (monolithic)

1.6.2. Cấu trúc phân lớp

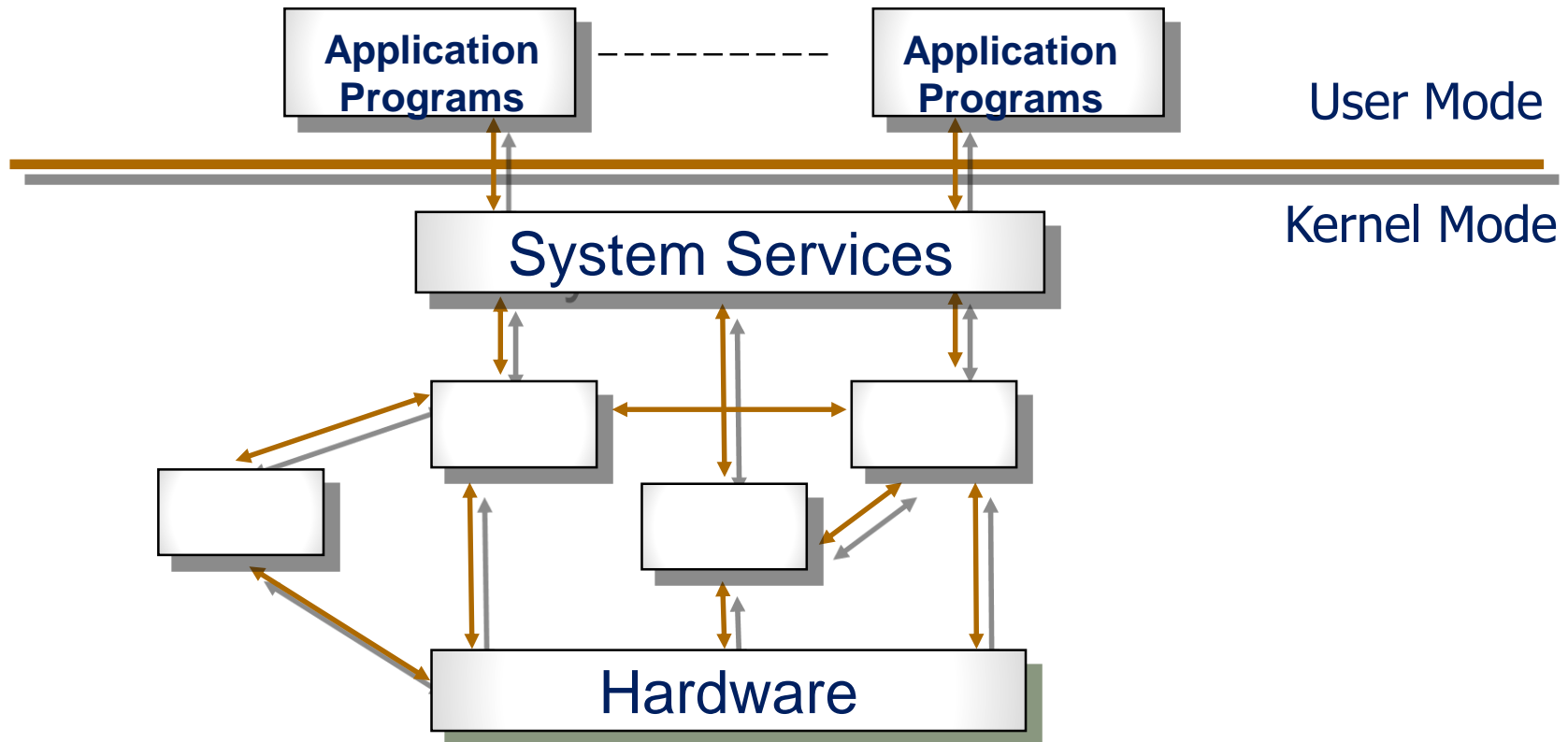
1.6.3. Máy ảo (Virtual machines)

1.6.4. Ngôn ngữ dùng để viết HĐH

1.6. Cấu trúc của Hệ điều hành

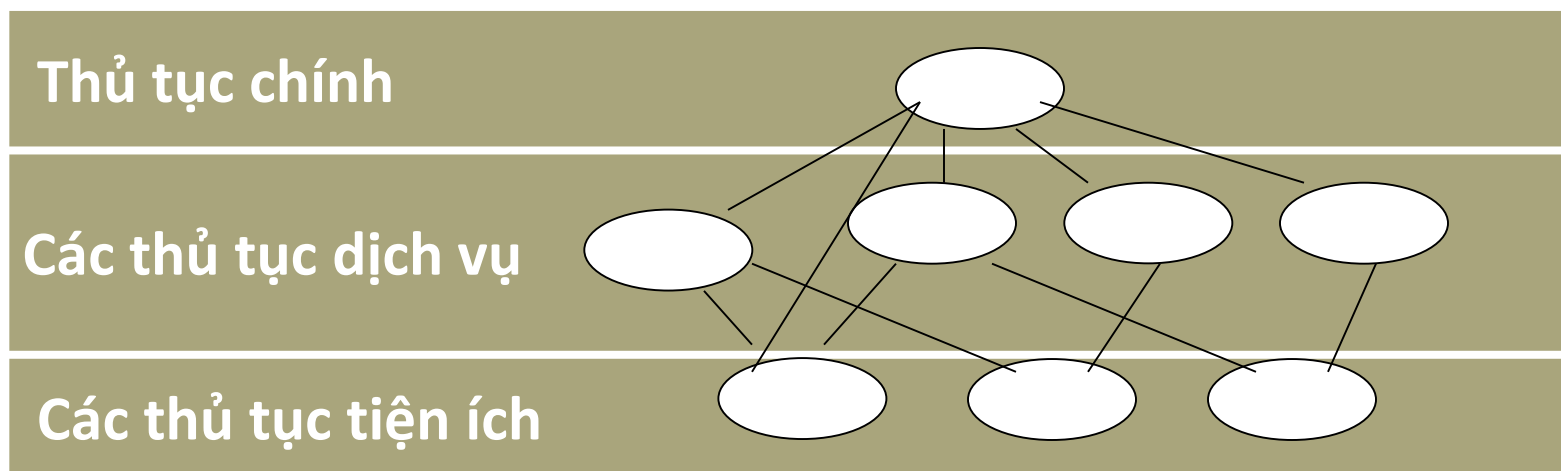
- HĐH là một hệ thống rất phức tạp, do đó cần phải được thiết kế cẩn thận để hoạt động đúng và dễ dàng sửa đổi.
- Phương pháp tốt nhất hiện nay là chia HĐH ra thành các thành phần nhỏ, mỗi thành phần đảm nhiệm một nhiệm vụ riêng bao gồm các chức năng được định nghĩa rõ ràng.
- Như vậy, cấu trúc logic của HĐH là qui tắc phân chia và kết nối các thành phần với nhau.

1.6.1. Cấu trúc đơn giản (monolithic)



1.6.1. Cấu trúc đơn giản (monolithic)

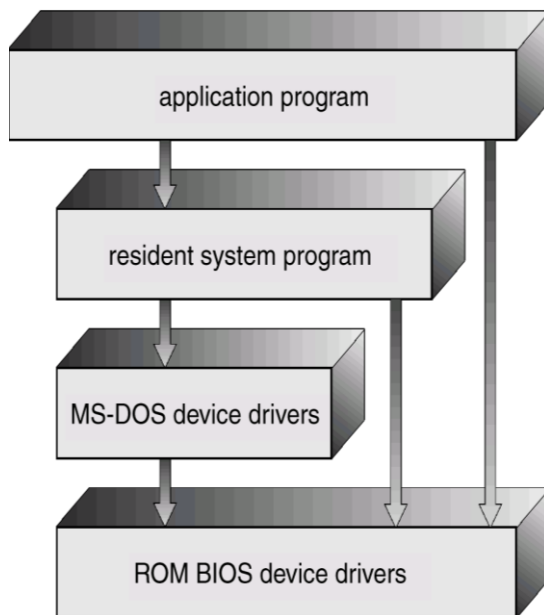
- Các HĐH theo cấu trúc đơn giản chia các thủ tục trong hệ thống thành 3 cấp:



- Thủ tục chính – chính là chương trình ứng dụng.
- Thủ tục dịch vụ hỗ trợ thủ tục chính gọi đến các thủ tục tiện ích
- Thủ tục tiện ích dùng để xử lý các lời gọi hệ thống.

1.6.1. Cấu trúc đơn giản (monolithic)

- Ví dụ: HĐH MS-DOS là một HĐH có cấu trúc đơn giản, xem như được phân lớp (layered)

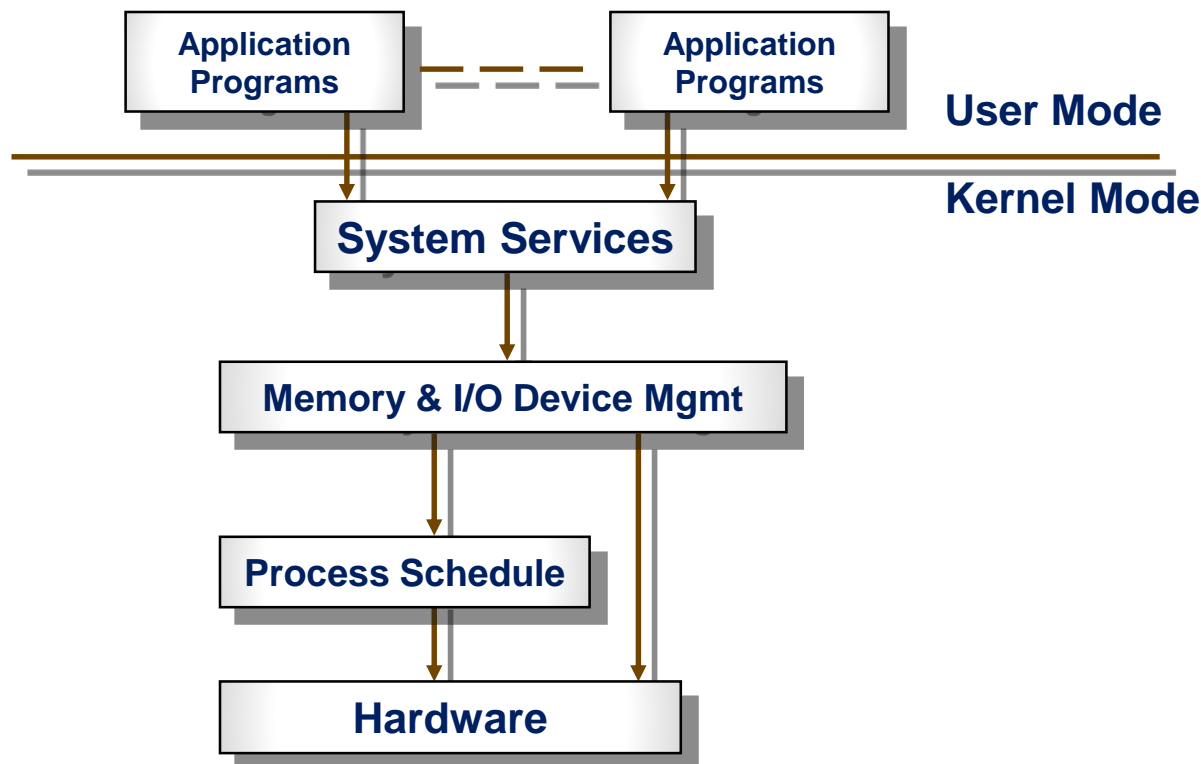


VD: Bộ diễn giải lệnh (command.com) là chương trình thường trú.

- Nhận xét: Với cấu trúc như trên chương trình ứng dụng có thể truy xuất trực tiếp đến phần cứng bằng cách gọi thủ tục tiện ích, điều này gây khó khăn cho HĐH trong việc kiểm soát và bảo vệ hệ thống.

1.6.2. Cấu trúc phân lớp

- Hệ thống được chia thành nhiều lớp, mỗi lớp được xây dựng dựa vào lớp bên dưới. Lớp dưới cùng là phần cứng, lớp trên cùng là giao diện với người sử dụng.



1.6.2. Cấu trúc phân lớp

Người sử dụng
Chương trình tiện ích (Shell, Compiler, System Libraries,...)
Thư viện chuẩn (Open, Read, Write, Close, ...)
Quản lý bộ nhớ chính, Lập lịch cho CPU, Quản lý file, thư mục, Quản lý bộ nhớ thứ cấp
Phần cứng (CPU, Memory, Disk,...)

- **Nhận xét:**

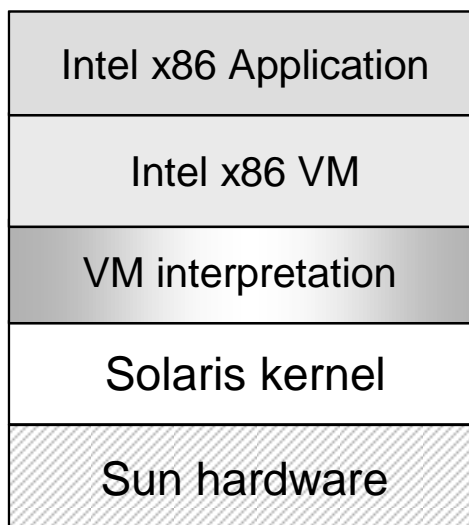
- Hệ thống có tính cấu trúc cao do đó dễ sửa đổi, phát hiện lỗi và kiểm nghiệm.
- Tuy nhiên HĐH có cấu trúc kiểu phân lớp sẽ hoạt động kém hiệu quả hơn do phải chi phí thời gian truyền thống dữ liệu giữa các lớp

1.6.3. Máy ảo (Virtual machines)

- Nhằm tăng cường tính độc lập cho chương trình ứng dụng đối với nền tảng phần cứng, người ta đưa ra khái niệm về máy ảo.
 - Máy ảo sẽ cung cấp một môi trường làm việc ảo cho các chương trình ứng dụng.
 - Máy ảo là bản sao chính xác các đặc tính phần phần cứng của một hệ thống máy tính tương ứng.
 - Các chương trình, HĐH khác nhau chạy trên các VM khác nhau, trong khi cùng nằm trên một máy thật.
 - Sử dụng máy ảo thì việc triển khai hệ thống sẽ dễ dàng hơn, chỉ cần thay đổi ở các VM.

1.6.3. Máy ảo (Virtual machines)

- VD: Làm thế nào để thực thi một chương trình MS-DOS trên một hệ thống Sun với hệ điều hành Solaris?
 - Tạo một máy ảo Intel bên trên hệ thống Sun và hệ điều hành Solaris
 - Các lệnh Intel (x86) được máy ảo Intel chuyển thành lệnh tương ứng của hệ thống Sun.



- Việc cài đặt phần mềm giả lập phần cứng là rất phức tạp. Việc áp dụng VM làm cho chương trình chạy chậm hơn.

1.6.4. Ngôn ngữ dùng để viết HĐH

- Trước đây HĐH được viết bằng ngôn ngữ assembly;
- Hiện nay HĐH được viết bằng ngôn ngữ bậc cao C.
- VD: Unix, Linux, Window
- Ưu điểm khi viết HĐH bằng ngôn ngữ bậc cao: viết nhanh hơn, cô đọng hơn, dễ hiểu và dễ gỡ rối. Dễ dàng hơn khi thay đổi nền tảng phần cứng.

1.7. Một số hệ điều hành phổ biến

- Microsoft: Windows ME, Windows 2000, Windows CE,...

- Linux

- BSD (FreeBSD, NetBSD)

- Sun: Solaris

- IBM: AIX

- Compaq: Tru64 Unix, OpenVMS

- Hewlett-Packard: HP-UX

- Apple: MacOS-8, MacOS-X

- Symbian: EPOC

- PalmOS

- QNX

- VXWorks

- LynxOS

- MVS, AS/400

- JVM, VMWare, DosBox, VirtualBox

Các dạng Unix

Thiết bị cầm tay

Các hệ nhúng thời gian thực

Máy ảo

1.8. Bài tập chương I

1. Mục đích chính của Hệ điều hành là gì?
2. Liệt kê các thành phần của tổ chức máy tính?
3. Lợi ích chính của Hệ điều hành đa chương là gì?
4. Khác biệt chính của máy tính mainframe và máy tính cá nhân là cái gì?
5. Phân biệt multiprocessor và distributed system.