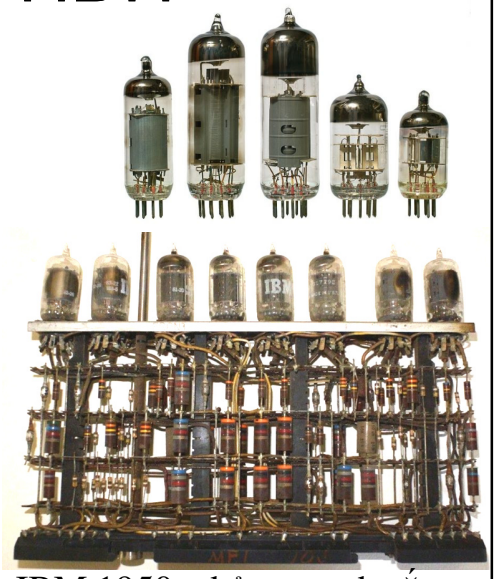
Nguyên lý Hệ điều hành

Bài mở đầu – Các khái niệm cơ bản

Lịch sử phát triển HĐH

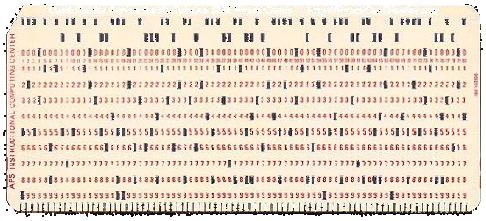
- Giai đoạn 45-55
  - Máy tính dùng bóng chân không và lập trình qua bảng mạch cắm
  - Do H.Aiken, J.V.Neuman, J.P Eckert, W. Mauchley, K.Zuse ...
  - Ngôn ngữ: mã máy
  - Chưa có khái niệm về HĐH



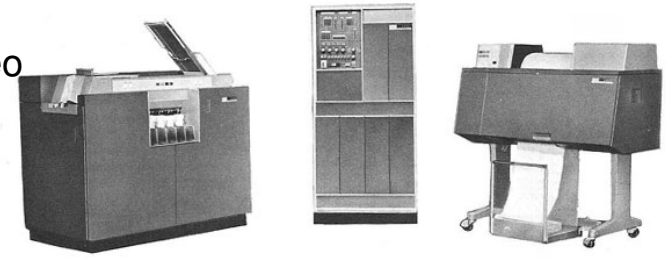
IBM 1950 – bảng mạch cắm

... Lịch sử phát triển HĐH

- Giai đoạn 55-65
  - Máy tính dùng mạch bán dẫn (vd: IBM 1401)
  - Chương trình nạp qua giấy đục lỗ, băng từ
  - HĐH: khả năng xử lý theo lô
  - Ngôn ngữ lập trình: hợp ngữ, fortran...



Punch card



Punch card reader IBM1401 Printer

... Lịch sử phát triển HĐH

- Giai đoạn 65-80
  - Máy tính dùng mạch tích hợp (IBM 360)
  - HĐH có khả năng xử lý đa nhiệm
  - Xuất hiện Unix, đầu 70s

... Lịch sử phát triển HĐH

- Unix
  - Multics (Multiplexed Information & Computing Services)
  - Unics (Uniplexed) => Unix
- Unix được viết trên C và hợp ngữ
  - C là ngôn ngữ cải tiến của B (BCPL)

Dennis Ritchie, Ken Thompson at PDP-11

... Lịch sử phát triển HĐH

- Giai đoạn 80-00
  - Máy tính dùng mạch tích hợp cỡ lớn (LSI, VLSI)
  - HĐH giành cho máy cá nhân
  - Giữa 80s: HĐH mạng và phân tán cho mạng các máy trạm nối với nhau.

Xu thế phát triển HĐH

- Giai đoạn trước ’80 - xu thế tập trung hóa tính toán
  - Luật Grosch: Hiệu suất tính toán tỷ lệ thuận với bình phương chi phí
- Giai đoạn ’80 – điện toán cá nhân
  - Tài nguyên tính toán được phân tán trên thiết bị cá nhân, có kết nối mạng
- Xu thế phát triển
  - Xử lý tập trung trong môi trường điện toán đám mây

Các thành phần của 1 hệ thống

Các ứng dụng
Compiler – Editor
Hệ điều hành
Ngôn ngữ máy
Micro program
Thiết bị vật lý

Khái niệm về HĐH

- Định nghĩa: là phần mềm hệ thống giúp người dùng quản lý và điều khiển các tài nguyên máy tính.
- Một hệ điều hành thường gồm 2 thành phần có sự độc lập với nhau
  - Hệ vỏ (shell), và
  - Hệ lõi (kernel)

Hệ vỏ

- Là trình thông dịch các lệnh của người dùng và thực hiện gọi các hàm thư viện hệ thống tương ứng của hệ lõi
- Có thể duy trì nhiều hệ vỏ cùng chạy 1 lúc

Hệ lõi

- Gồm mã phần cốt lõi của HĐH, nơi thực hiện thao tác, quản lý trực tiếp với các tài nguyên hệ thống. Hệ lõi nhận yêu cầu từ hệ vỏ, thực hiện công việc và trả lại kết quả cho hệ vỏ
- Thiết kế độc lập hệ vỏ
- Chỉ tồn tại 1 hệ lõi được chạy

Tiến trình

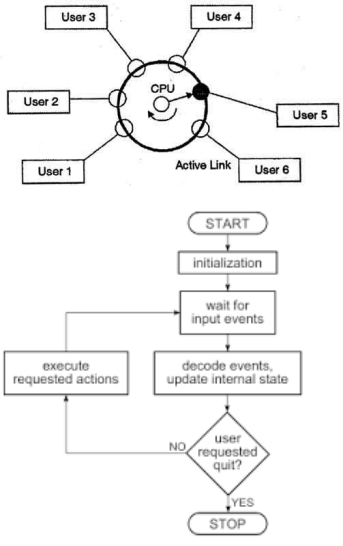
- Tiến trình (process): là một chương trình đang chạy trong bộ nhớ được cấp phát tài nguyên và quản lý bởi HĐH. Mỗi tiến trình được cấp một số hiệu (PID) để quản lý.
- Một tiến trình có thể bị dừng tạm thời và HĐH chuyển qua chạy 1 tiến trình khác
- Một tiến trình có thể sinh ra tiến trình con
- Các tiến trình trao đổi với nhau thông qua signal (~ software interrupt)
- HĐH lưu bảng các tiến trình để nhớ trạng thái hoạt động
- Cơ chế dùng signal cho phép các tiến trình làm việc khác trong khi chờ

Đơn nhiệm và đa nhiệm

- HĐH đơn nhiệm: chỉ một tiến trình được phép chạy. Cần phải kết thúc tiến trình này để chạy tiến trình khác.

...Đơn nhiệm và đa nhiệm

- HĐH đa nhiệm: có thể tồn tại nhiều tiến trình được chạy nằm trong bộ nhớ.
  - Đa nhiệm ưu tiên: Mỗi tiến trình được lần lượt phân chia một khoảng thời gian CPU nhất định và tài nguyên hệ thống để chạy. Nếu hết khoảng thời gian đó, tiến trình phải xếp hàng chờ lần tiếp theo.
  - Đa nhiệm không ưu tiên: mỗi tiến trình được lần lượt giao quyền điều khiển CPU và tài nguyên và phải tự nhả ra khi xử lý xong 1 sự kiện.
- HĐH đa người dùng: tồn tại nhiều tiến trình thuộc nhiều người dùng khác nhau cùng chạy trên hệ thống



Kiến trúc một HĐH

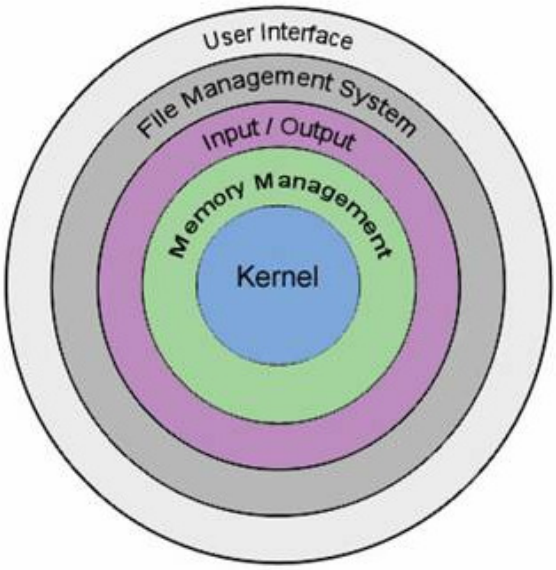
- Kiến trúc đơn tầng
  - HĐH được viết như một tập các hàm dịch vụ, gọi được lẫn nhau
  - Chế độ chạy dựa vào khả năng chạy của CPU trên 2 mức user và kernel.
- Quy trình:
  - Chương trình người dùng gọi lệnh bẫy hệ thống; CPU chuyển qua chế độ kernel
  - HĐH kiểm tra bảng thông số và gọi hàm dịch vụ tương ứng
  - Kết quả và điều khiển được trả về chương trình người dùng; CPU chuyển lại về user mode

System calls

- Dùng để yêu cầu HĐH thực hiện một nhiệm vụ
- Quy trình thực hiện của hàm thư viện HT:
  - Lưu tham số vào 1 vùng bộ nhớ, thanh ghi
  - Gọi lệnh bẫy HĐH thực hiện nhiệm vụ
  - HĐH đặt kết quả vào thanh ghi/bộ nhớ và gọi lệnh kết thúc bẫy, quay trở lại hàm thư viện
  - Hàm thư viện lấy kết quả, trả lại cho người gọi

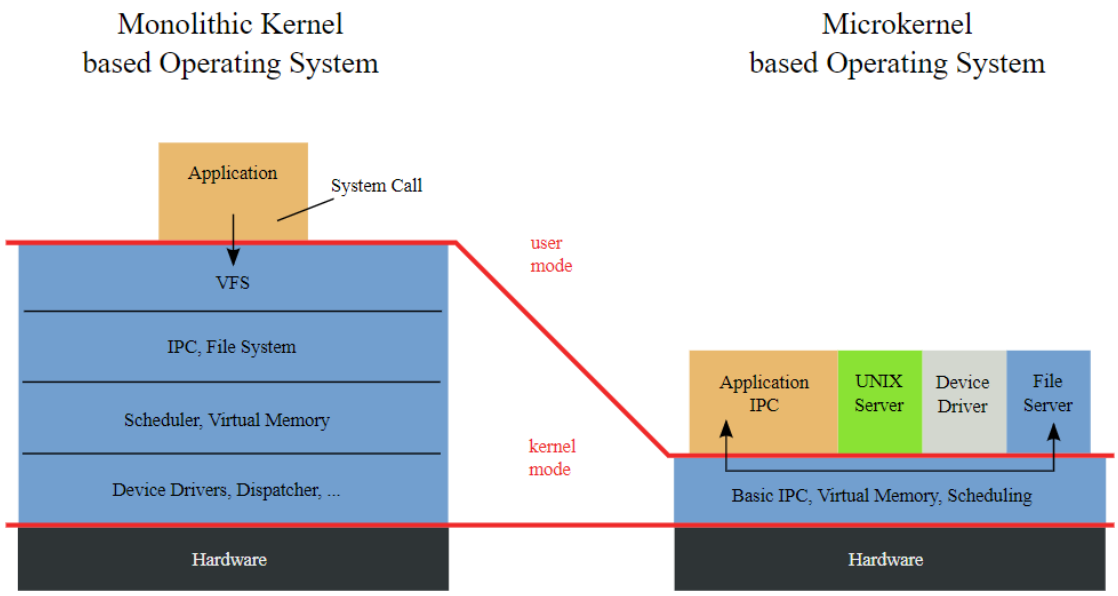
Kiến trúc một HĐH

- Kiến trúc phân tầng:
  - Tầng 1: phân phối CPU, quản lý đa nhiệm
  - Tầng 2: Quản lý bộ nhớ
  - Tầng 3: Quản lý vào/ra; quản lý liên lạc giữa các tiến trình
  - Tầng 4: Quản lý tệp
  - Tầng 5: Quản lý giao diện người dùng



Mô hình monolithic và micro kernel

- Mô hình nguyên khối (monolithic)
  - Các chức năng của 5 tầng kiến trúc đều được cài đặt trong nhân hệ điều hành và được thực hiện ở chế độ kernel mode
- Mô hình vi nhân (micro kernel)
  - Nhân hệ điều hành chỉ cung cấp các dịch vụ quản lý tiến trình, bộ nhớ cơ bản
  - Các dịch vụ quản lý tệp, liên lạc giữa các tiến trình, điều khiển thiết bị vv.. đều được cài đặt dưới dạng service và chạy ở chế độ user mode
  - Các dịch vụ mở rộng được gọi qua mô hình client-server
  - Là kiến trúc phù hợp với hệ điều hành phân tán vì dễ dàng hơn trong việc chia sẻ công việc giữa các máy



Nguyên lý Hệ điều hành

Khung chương trình, tài liệu, nội dung thực hành

Khung chương trình

1. Tiến trình
2. Quản lý bộ nhớ
3. Hệ thống tệp
4. Kiến trúc vào/ra
5. Vấn đề bế tắc
6. Sơ bộ về các vấn đề trong hệ thống phân tán

Tài liệu

- Link tài liệu: mã QR
- Google classroom

  A4: gz2doix A7: llcn4jq

  A6: sxuhsuq A3: r4litle
- Sách tham khảo: Modern Operating System, A. Tanenbaum 4th Edition

Thực hành

- Ngôn ngữ: c cơ bản / Linux
- Các lựa chọn cài đặt:
  - Windows 10: cài app Ubuntu và chạy Bash shell, chạy
    - sudo apt-get update
    - Sudo apt-get install gcc
  - Cài vmware và cài ubuntu
  - Cài cygwin terminal theo hướng dẫn trong link môn học

Nội dung thực hành

- Lập trình luồng
- Sử dụng semaphore giải quyết cạnh tranh tài nguyên
- Quản lý bộ nhớ: cấp phát/giải phóng, QL bộ nhớ ảo
- Hệ thống tệp: xây dựng hệ thống FAT, inodes
- Deadlock: thuật toán nhà băng