

ỦY BAN NHÂN DÂN THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG CAO ĐẲNG KỸ THUẬT LÝ TỰ TRỌNG TP.HCM

ĐỀ TÀI SÁNG KIẾN KINH NGHIỆM

**“Xây dựng bộ câu hỏi môn học Nguyên lý hệ điều hành bậc cao đẳng
nghề ngành Công nghệ thông tin”**

Biên soạn: Dương Ngọc Vọng

TP.HỒ CHÍ MINH, THÁNG 01/2016

MỞ ĐẦU

Nguyên lý hệ điều hành là cách thức hệ điều hành quản lý thiết bị phần cứng từ tổ chức quản lý hệ tập tin, quản lý bộ nhớ và quản lý các tiến trình cũng như cách thức xử lý thông tin. Hệ điều hành còn là cầu nối giúp người sử dụng tương tác và kết nối với thiết bị phần cứng. Nắm được kiến thức về nguyên lý hệ điều hành một cách tổng thể sẽ giúp ích rất nhiều khi tìm hiểu và vận hành các hệ điều hành thương mại đang phổ biến rộng rãi hiện nay như Window, Linux, Adroid... Để sinh viên dễ dàng tiếp cận và khai thác hiệu quả thì việc xây dựng bộ câu hỏi cho môn học nguyên lý hệ điều hành giúp cho người học hệ thống được các kiến thức trọng tâm của môn học. Trong quá trình biên soạn nhằm củng cố kiến thức và làm đáp án không tránh khỏi sự thiếu sót mong được góp ý để chỉnh sửa được hoàn thiện hơn.

I- NỘI DUNG

Câu 1: Cho một Volume có kích thước 2.055.168 bytes, kích thước sector 512 bytes, vùng SYSTEM chiếm 11 sector, mỗi cluster chiếm 4 sectors, cluster đầu tiên được đánh chỉ số là 2; thì phân bố cluster trên Volume sẽ được biểu diễn như thế nào?

Câu 2: *Nêu công dụng và giải thích nội dung của 3 tập tin hệ thống cho dưới đây?*

a- /etc/passwd

/etc/passwd

- root:x:0:0:root:/root:/bin/bash
- bin:x:1:1:bin:/bin:/sbin/nologin
- daemon:x:2:2:daemon:/sbin:/sbin/nologin
- adm:x:3:4:adm:/var/adm:/sbin/nologin
- lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
- ...

b- /etc/group

/etc/group

- root:x:0:root
- bin:x:1:root,bin,daemon
- daemon:x:2:root,bin,daemon
- sys:x:3:root,bin,adm
- adm:x:4:root,adm,daemon
- ...
- mysql:x:27:
- postgres:x:26:
- hv:x:500:
- ...

c- /etc/shadow

/etc/shadow

- root:\$1\$s3fAJLqH\$miwuH9XjHQeFuVcPsq/.2
13620:0:99999:7:::
- bin:!:13620:0:99999:7:::
- mysql:!:13620:0:99999:7:::
- postgres:!:13620:0:99999:7:::
- hv:\$1\$8tX7pMdC\$cDM9/UhM0SKa4wYI9S2JC
:13620:0:99999:7:::

Câu 3: Hệ điều hành là gì? Nêu sự khác biệt giữa HĐH và các phần mềm thương mại hiện nay?

Câu 4 :

Cho nhóm tập lệnh sau : cls, ls, dir, cd, rd, copy con, cp, mkdir, cat, type, rd, del, rm, copy, gzip, gunzip

Câu a: Xác định nhóm lệnh trên được sử dụng trong hệ điều hành thương mại nào hiện nay ? (1 đ)

Câu b: Trình bày cú pháp và công dụng của từng lệnh trên ? (2 đ)

Câu 5 :

Cho bảng tiến trình sau :

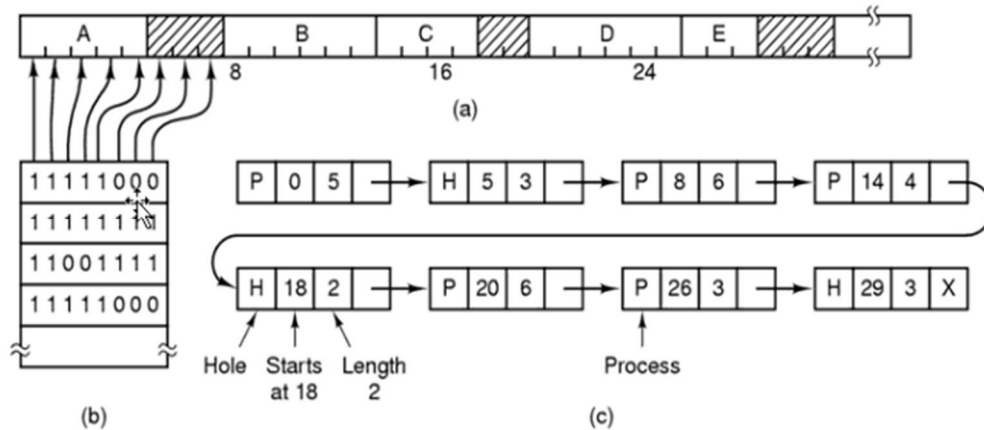
Tiến trình	Thời gian vào	Thời gian xử lý
P0	0/0	27/23
P1	1/0	19/17
P2	2/0	16/22
P3	3/0	7/18
P4	4/0	6/9

Câu a: Xác định thời gian chờ trung bình trong chiến lược điều phối RR với Quantum=3 và FCFS ? (2 đ)

Câu b: Xác định thời gian chờ trung bình trong chiến lược điều phối tiến trình với SJF ? (2 đ)

Câu 6 :

Cho sơ đồ quản lý bộ nhớ trong tiến trình cấp phát động bằng danh sách liên kết sau :



Hãy giải thích ngắn gọn ?

Câu 7 :

Cho 128 byte đầu tiên của 1 bảng Boot Sector với nội dung như sau :

EB	3C	90	4D	53	57	49	4E	34	2E	31	00	02	10	01	00
02	00	02	00	00	F8	FF	00	3F	00	FF	00	3F	00	00	00
C2	EE	0F	00	80	00	29	DE	1C	49	15	20	20	20	20	20
20	20	20	20	20	20	46	41	54	31	36	20	20	20	33	C9
8E	D1	BC	F0	7B	8E	D9	B8	00	20	8E	C0	FC	BD	00	7C
38	4E	24	7D	24	8B	C1	99	E8	3C	01	72	1C	83	EB	3A
66	A1	1C	7C	26	66	3B	07	26	8A	57	FC	75	06	80	CA
02	88	56	02	80	C3	10	73	EB	33	C9	8A	46	10	98	F7

Xác định thông tin của bảng Boot Sector trên ?

Câu 8: Vẽ sơ đồ và nêu tóm tắt hệ thống các thành phần cơ bản của máy tính?

Câu 9: Nêu công dụng của các thanh ghi AX, BX, CX và DX?

Câu 10: Nêu 2 công dụng chính của hệ điều hành?

Câu 11: Nêu một số cách phân loại hệ điều hành?

Câu 12: Hệ điều hành đơn nhiệm là gì? Đa nhiệm là gì? Nêu một số hệ điều hành thông dụng hiện nay?

Câu 13: Nêu các chức năng chính và các thành phần của hệ điều hành?

Câu 14: Nêu quá trình khởi động và kiểm tra của một thông máy tính?

Câu 15: Bus là gì? Công dụng của Bus địa chỉ và Bus dữ liệu?

Câu 16: Nêu Một số nhận sắc về sự khác biệt giữa HĐH và các phần mềm khác?

Câu 17: Nêu một số khái niệm về hệ điều hành?

Câu 18: Trình bày cấu trúc của hệ điều hành?

Câu 19: Nêu hoạt động của hệ điều hành nhìn góc độ bên trong?

Câu 20: Trình bày lịch sử phát triển của hệ điều hành?

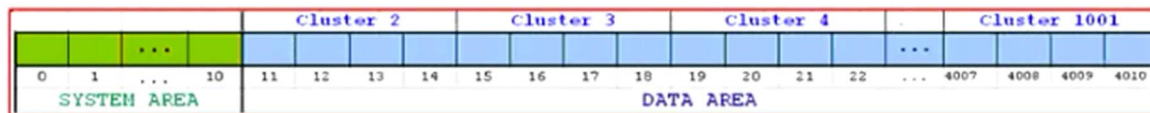
Câu 21: Nêu giải pháp cần thiết cho việc quản lý hệ thống tập tin?

- Câu 22: Nêu khái niệm tập tin và thư mục?
- Câu 23: Trình bày hiểu biết của Anh/Chị về khái niệm Volume?
- Câu 24: Trình bày thiết kế hệ thống lưu trữ tập tin?
- Câu 25: Trình bày thiết kế hệ thống lưu trữ thư mục?
- Câu 26: Nêu cách thức tổ chức hệ thống tập tin trên Volume?
- Câu 27: Trình bày khái niệm Cluster? Nêu tính cấp thiết cho việc dùng Cluster?
- Câu 28: Trình bày hình thức tổ chức lưu trữ bằng Cluster?
- Câu 29: Trình bày các yếu tố để xác định kích thước Cluster?
- Câu 30: Bảng Cluster là gì? Nêu lý do phát sinh?
- Câu 31: Nêu hình thức tổ chức quản lý bằng Cluster? Cho ví dụ?
- Câu 32: Nêu cách Quản lý chuỗi các cluster của tập tin bằng xâu(chuỗi)?
- Câu 33: Nêu cách Quản lý chuỗi các cluster của tập tin bằng xâu kết hợp với chỉ mục(index)?
- Câu 34: Trình bày cách Quản lý chuỗi các Cluster của tập tin bằng cấu trúc cây?
- Câu 35: Trình bày cách tổ chức tập tin hệ thống I-NODE đối với hệ điều hành UNIX?
- Câu 36: Nêu khái niệm bảng thư mục? cho ví dụ minh họa?
- Câu 37: Boot Sector là gì? Mô tả cấu trúc bảng boot sector?
- Câu 38: Bảng FAT là gì? Kể tên một số loại FAT?
- Câu 39: Trình bày khái niệm về tiến trình?
- Câu 40: Nêu các trạng thái của tiến trình?
- Câu 41: Trình bày khái niệm tiêu trình?
- Câu 42: Hàng đợi tiến trình là gì?
- Câu 43: Nêu các tiêu chuẩn điều phối tiến trình?
- Câu 44: Trình bày chiến lược điều phối tiến trình bằng FIFO(First In First Out)? Cho ví dụ?
- Câu 45: Trình bày điều phối tiến trình bằng FCFS (First Come – First Served)?
- Câu 46: Trình bày chiến lược điều phối tiến trình bằng ROUND-ROBIN?
- Câu 47: Trình bày điều phối tiến trình bằng giải thuật SJF
- Câu 48: Trình bày cơ chế điều phối tiến trình trung dụng và không trung dụng CPU?
- Câu 49: Các đặc điểm của hệ thống phân tán?
- Câu 50: Mục tiêu thiết kế hệ điều hành phân tán là gì?
- Câu 51: Trạng thái deadlock là gì?
- Câu 52: Cơ chế nào để ngăn deadlock?
- Câu 53: Cơ chế phân trang là gì?
- Câu 54: Cơ chế phân đoạn là gì?
- Câu 55: Trình bày trạng thái quản lý tiến trình trong hệ đa chương đối với phân vùng động?
- Câu 56: Trình bày Cơ chế quản lý vùng nhớ trống bằng danh sách liên kết? Vẽ sơ đồ minh họa?
- Câu 57: Swapping là gì? Anh/chị nêu một trường hợp cụ thể cho cơ chế swapping này đối với hệ điều hành Linux?
- Câu 58: Trình bày tóm lược cách sử dụng bộ nhớ của hệ điều hành?
- Câu 59: Trình bày cách quản lý bộ nhớ trong hệ điều hành Linux?
- Câu 60: Trình bày các dịch vụ chính của hệ điều hành?

II- PHẦN ĐÁP ÁN

“Biên soạn bộ đề thi tự luận cho học phần Nguyên lý hệ điều hành bậc cao đẳng nghề ngành Công nghệ thông tin”

Câu 1 : (2.5 điểm): Sơ đồ phân bố Volume được biểu diễn bằng hình vẽ như sau:



Câu 2: (5 điểm) :

Đáp án

Câu 3a: (2 điểm)

Tập tin /etc/passwd: mô tả thông tin về người dùng

```
/etc/passwd
■ root:x:0:0:root:/root:/bin/bash
■ bin:x:1:1:bin:/bin:/sbin/nologin
■ daemon:x:2:2:daemon:/sbin:/sbin/nologin
■ adm:x:3:4:adm:/var/adm:/sbin/nologin
■ lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
■ ...
```

+Mỗi dòng trong tập tin /etc/passwd là 1 user

+Mỗi dòng có 7 trường, mỗi trường cách nhau bằng dấu hai chấm ‘:’

Trường 1: Mô tả tên user (example : root)

Trường 2 : user đang hoạt động được đánh dấu x, user đang bị tam khóa đánh dấu *

Trường 3 : Cho biết user ID của user

Trường 4 : Cho biết về group ID của user

Trường 5 : Cho biết thông tin cá nhân của user

Trường 6 : Cho biết Home Directory của user

Trường 7 : Shell thực hiện khi đăng ký tên user

Câu 3b: (1.5 điểm)

Tập tin etc/group: mô tả thông tin về tên nhóm

```
/etc/group
■ root:x:0:root
■ bin:x:1:root,bin,daemon
■ daemon:x:2:root,bin,daemon
■ sys:x:3:root,bin,adm
■ adm:x:4:root,adm,daemon
■ ...
■ mysql:x:27:
■ postgres:x:26:
■ hv:x:500:
■ ...
```

Trường 1 : Thông tin về Group (ex: nhóm root)

Trường 2 : Group hoạt động đánh dấu x, tạm khóa đánh dấu *

Trường 3 : GroupID của nhóm

Trường 4 : Danh sách các user thuộc nhóm

Câu 3c: (1.5 điểm)

Tập tin etc/shadow: Chứa thông tin user và mật khẩu đã được mã hóa

```
/etc/shadow
■ root:$1$s3fAJLqH$miwuH9XjHQeFuVcPsql/.2
13620:0:99999:7:::
■ bin:!:13620:0:99999:7:::
■ mysql:!:13620:0:99999:7:::
■ postgres:!:13620:0:99999:7:::
■ hv:$1$8tX7pMdc$cDM9/UhM0SKa4wYI9S2JC
:13620:0:99999:7:::
```

Trường 1: Chỉ username phải khớp với username trong file /etc/passwd

Trường 2: Chỉ mật khẩu đã được mã hóa (dấu * cho biết tài khoản đã bị vô hiệu hóa)

Trường 3: Số ngày tính từ ngày 1/1/1970 đến ngày thay đổi mật khẩu sau cùng

Trường 4: Số ngày cần thay đổi mật khẩu khi lần đầu sử dụng (số 0 : cho phép thay đổi bất kỳ lúc nào)

Trường 5: Cho biết thời gian tồn tại của mật khẩu, trong đó giá trị 99999 là không cần thay đổi

Câu 3: Nêu khái niệm về hệ điều hành? Cho một vài nhận xét về sự khác biệt giữa HĐH và các phần mềm khác?

Đáp án:

- Khái niệm HĐH :

+ Là một hệ thống các chương trình để điều khiển máy tính. Hệ điều hành gắn liền với máy tính là “linh hồn” của máy tính.

+ Hệ điều hành là một hệ thống các chương trình chạy trên máy tính, dùng để điều hành quản lý các thiết bị phần cứng và các tài nguyên phần mềm trên máy tính.

+HĐH đứng trung gian giữa người sử dụng và phần cứng máy tính, nó tạo ra một môi trường để người sử dụng có thể dùng máy tính tiện lợi, dễ dàng và hiệu quả hơn.

+HĐH là 1 chương trình đặc biệt: Tự động thực hiện khi khởi động máy tính, và khi thoát tự động tắt máy. Phần nhiều (hầu hết) các chương trình khác hoạt động được là nhờ đến sự hỗ trợ của HĐH.

- Một số nhận xét về sự khác biệt giữa HĐH và các phần mềm khác: (1 điểm)

+Khả năng tự hoạt động ngay sau khi bật máy

+Tác động đến máy tính khi chương trình kết thúc

+Mức độ cần thiết cho sự hoạt động tối thiểu của máy tính

+Khả năng điều khiển phần cứng

+So với các phần mềm khác thì phần mềm HĐH được thiết kế và lập trình phức tạp hơn nhiều.

+Phần mềm đầu tiên phải cài đặt vào hệ thống máy tính là phần mềm HĐH

+Trên hệ thống máy tính có thể có nhiều HĐH nhưng tại một thời điểm chỉ có một HĐH hoạt động.

+Là phần mềm bắt buộc phải có, trong khi các phần mềm khác có thể có hoặc không tùy vào nhu cầu của người sử dụng.

Câu 4 :

Câu 4a :

Hệ điều hành DOS/Windows	Hệ điều hành Unit/Linux
Cls,dir,cd,rd, copy con, type, copy, del	Ls, cd, cp, mkdir, rm, gzip, gunzip

Câu 4b:

- Hệ điều hành DOS/Windows:

> Cls : lệnh xóa màn hình

> Cd : lệnh chuyển thư mục làm việc

>Rd /path/Folder_name : lệnh xóa

>copy con /path/file_name :lệnh tập tập tin

>del /path/file_name : lệnh xóa tập tin

>Type /path/File_name : lệnh xem nội dung tập tin

- Hệ điều hành Unit/Linux :

>ls -a : hiển thị tất cả các tập tin kể cả tập tin ẩn

>ls -x : hiển thị trên nhiều cột

>ls -l : hiển thị chi tiết thông tin chứa trong thư mục

* **cat > file_name** : Sẽ tạo mới nội dung file. Nếu file tạo mới trùng tên với file cũ → nội dung file cũ sẽ bị ghi đè. Kết thúc soạn thảo nội dung nhấn tổ hợp phím Ctrl + d

* **cat >> file_name** : Cho phép nối thêm nội dung vào cuối file_name. Kết thúc nhập nội dung nối file → nhấn tổ hợp phím Ctrl + d

>cp </path/source_file> </path/destination_file> : Lệnh copy thư mục/tập tin

Ex : \$cp /home/*. * /home/Source

>rm <path/file_name> lệnh xóa tập tin

Ex : \$ rm /bin/a2.dat

>mkdir <[path]/Folder_name> : lệnh tạo thư mục

Ex : \$mkdir /home/AAA

>rm [/path]/File name : lệnh xóa tập tin

Ex: \$mkdir /home/AAA

>cd [/path]/Folder_name : lệnh chuyển thư mục làm việc
việc (thư mục hiện hành)

>gzip [/path]/File_name → sau khi nén File_name ở dạng File_name.gz : lệnh nén tập tin/thư mục

>gunzip [/path]/File_name.gz : lệnh giải nén tập tin thư mục

Câu 5 :

Câu 5a :

- Thời gian chờ trung bình theo chiến lược điều phối RR là 38,8 (ms)
- Thời gian chờ trung bình theo chiến lược điều phối FCFS là 38,8 (ms)

Câu 5b :

- Thời gian chờ trung bình với chiến lược điều phối SJF là 29 (ms)

Câu 6 :

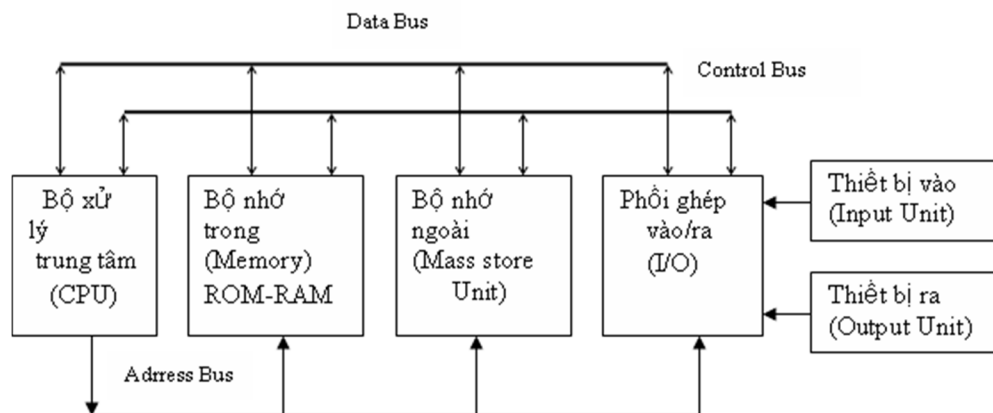
Tiền trình P bắt đầu tại bit 0, chiếm 5 bit và là giá trị lưu chỉ số của phần tử kế tiếp. Từ đó, ta có thể suy ra phần tử kế tiếp cho đến khi gặp phần tử kết thúc (H,29,3,X) trong đó X xem như là giá trị NULL

Câu 7 :

- Số byte trên mỗi sector =512d
- Kích thước Cluster=16d
- Số sector trước vùng FAT =1d
- Số bảng FAT =2d (0.5 đ)
- Kích thước bảng FAT=255d
- Kích thước Volume=1044162d

Câu 8:

Sơ đồ khối tổng quát của máy tính số



Trong sơ đồ này, các khối chức năng chính của máy tính số gồm:

- Khối xử lý trung tâm (central processing unit, CPU),
- Bộ nhớ trong (memory), như RAM, ROM
- Bộ nhớ ngoài, như các loại ổ đĩa, băng từ
- Khối phối ghép với các thiết bị ngoại vi (vào/ra)

- Các bộ phận đầu vào, như bàn phím, chuột, máy quét

- Các bộ phận đầu ra, như màn hình, máy in

Bốn khối chức năng đầu liên hệ với nhau thông qua tập các đường dây để truyền tín hiệu, gọi chung là *bus hệ thống*. Bus hệ thống bao gồm 3 bus thành phần; ứng với các tín hiệu xác lập địa chỉ từ CPU đến các đơn vị thành phần ta có bus địa chỉ; với các dữ liệu được liên hệ giữa các khối qua bus dữ liệu (data bus); các tín hiệu điều khiển bao gồm các lệnh, các đáp ứng, các trạng thái của các khối được xác lập qua bus điều khiển.

Sự khác biệt quan trọng nhất của các hệ máy tính là kích thước và tốc độ, các máy tính nhỏ hơn và nhanh, mạnh hơn theo từng năm. Sự phát triển không ngừng của các thế hệ máy tính nhờ vào hai yếu tố quan trọng, đó là sự phát triển của công nghệ chế tạo IC và công nghệ chế tạo bộ nhớ.

Câu 9:

- AX (accumulator) rộng 16 bit, được chia làm hai phần: 1 byte cao AH và 1 byte thấp AL. Đây là thanh ghi quan trọng nhất và chuyên được dùng để chứa kết quả các thao tác lệnh. Cả ba cách viết AX, AH, AL đều có thể sử dụng như những thanh ghi riêng biệt.

- BX (base) thanh ghi cơ sở, rộng 16 bit, cũng được chia ra làm BH và BL. Đây là thanh ghi thường dùng chứa địa chỉ cơ sở của một bảng dùng trong lệnh XLAT, Cả ba cách viết BX, BH, BL đều có thể sử dụng như những thanh ghi riêng biệt.

- CX (count) bộ đếm, rộng 16 bit. Được chia ra làm CH và CL. Thanh ghi CX được dùng để chứa số lần lặp trong trường hợp các lệnh LOOP. Thanh ghi thấp CL được dùng để chứa (nhớ) số lần quay hoặc dịch của các lệnh quay (rotate) và dịch (shift).

- DX (data) thanh ghi dữ liệu, rộng 16 bit. Thanh ghi này cùng thanh ghi AX tham gia vào các thao tác của phép nhân hoặc chia các số 16 bit. DX còn dùng để chứa địa chỉ 16 bit của các cổng cứng (dài hơn 8 bit) trong các lệnh truy nhập các cổng ngoại vi (I/O port).

Câu 10:

- Hệ điều hành là một phần mềm chạy trên máy tính, dùng để điều hành, quản lý các thiết bị phần cứng và các tài nguyên phần mềm trên máy tính.
- Hệ điều hành đóng vai trò trung gian trong việc giao tiếp giữa người sử dụng và phần cứng máy tính, cung cấp một môi trường cho phép người sử dụng phát triển và thực hiện các ứng dụng của họ một cách dễ dàng.

Câu 11:

- Hệ điều hành dành cho máy MainFrame
- Hệ điều hành dành cho máy Server
- Hệ điều hành dành cho máy nhiều CPU
- Hệ điều hành dành cho máy tính cá nhân (PC)
- Hệ điều hành dành cho máy PDA (Embedded OS - hệ điều hành nhúng)
- Hệ điều hành dành cho máy chuyên biệt
- Hệ điều hành dành cho thẻ chip (SmartCard)

Câu 12:

+Hệ điều hành *đơn nhiệm* một người dùng (các chương trình phải được thực hiện lần lượt, ví dụ: HĐH MS-DOS)

+Hệ điều hành *đa nhiệm* một người dùng (tức là có thể thực hiện đồng thời nhiều chương trình, ví dụ Windows 95).

+Hệ điều hành *đa nhiệm* nhiều người dùng (Windows XP, Windows 7, Windows 8)

+Một số hệ điều hành thông dụng Android. Windows, Mac OS và Mac OS X

Câu 13

Chức năng chính yếu của hệ điều hành

- Tổ chức giao tiếp giữa người dùng và hệ thống;
- Cung cấp tài nguyên (bộ nhớ, các thiết bị ngoại vi,...) cho các chương trình và tổ chức thực hiện các chương trình đó;

- Tổ chức lưu trữ thông tin trên bộ nhớ ngoài, cung cấp các công cụ để tìm kiếm và truy cập thông tin;
- Kiểm tra và hỗ trợ bằng phần mềm cho các thiết bị ngoại vi (chuột, bàn phím, màn hình, đĩa CD,...) để có thể khai thác chúng một cách thuận tiện và hiệu quả;
- Cung cấp các dịch vụ tiện ích hệ thống (làm việc với đĩa, truy cập mạng,...).

Các thành phần của hệ điều hành

- +Hệ thống quản lý tiến trình
- +Hệ thống quản lý bộ nhớ
- +Hệ thống quản lý nhập xuất
- +Hệ thống quản lý tập tin
- +Hệ thống bảo vệ
- +Hệ thống dịch lệnh
- +Quản lý mạng

Câu 14:

QUÁ TRÌNH KHỞI ĐỘNG VÀ KIỂM TRA CỦA MÁY TÍNH

- Quá trình khởi động và kiểm tra của máy tính diễn ra ngay sau khi bạn bấm công tắc mở nguồn, khi mà màn hình chưa có gì cả là lúc một loạt quá trình đã được thực hiện bởi chương trình POST máy do BIOS thực hiện.
- Hầu hết các hư hỏng của Mainboard đều biểu hiện ở lúc khởi động, vì vậy nếu bạn nắm chắc được quá trình khởi động của máy thì bạn có thể dễ dàng xác định được nguyên nhân của mỗi sự cố.

Các bước trong quá trình khởi động máy tính (sau khi bật công tắc)

1. Bật công tắc, nguồn chính hoạt động cung cấp cho Mainboard các điện áp chính 12V, 5V và 3.3V
2. Mạch VRM cấp nguồn VCORE cho CPU đồng thời báo tín hiệu VRM_GD (VRM_Good) đến Chipset nam
3. Mạch tạo xung Clock (Clocking) hoạt động, cung cấp cho các thành phần trên Main xung Clock để hoạt động
4. Khi có Vcc, có xung Clock IC-SIO hoạt động.
5. IC-SIO tạo tín hiệu Reset để khởi động Chipset nam
6. Chipset nam hoạt động
7. Nếu có tín hiệu VRM_GD thì Chipset nam tạo tín hiệu Reset hệ thống.
8. Chipset bắc hoạt động
9. Chipset bắc tạo ra tín hiệu Reset CPU
10. CPU hoạt động
11. CPU phát tín hiệu truy cập ROM để nạp chương trình BIOS
12. Chương trình BIOS kiểm tra bộ nhớ RAM
13. Chương trình BIOS kiểm tra Card Video
14. BIOS cho nạp bản lưu cấu hình máy trong RAM CMOS
15. Kiểm tra các cổng và các ổ đĩa theo thiết lập trong CMOS
16. Khởi động ổ cứng và nạp hệ điều hành từ ổ cứng lên RAM
17. Quá trình khởi động máy tính sau khi bật công tắc

Câu 15:

Toàn bộ mạch điều khiển bên trong các máy tính thuộc dòng PC được liên kết với nhau nhờ một mạng lưới gọi là bus. Bus đơn thuần là một đường dùng chung, nằm trên bảng mạch chính mà mọi thành phần điều khiển máy tính được nối vào. Khi dữ liệu được truyền từ phần này sang phần khác nó di chuyển dọc theo con đường chung này để đến đích.

Bus địa chỉ: Đối với hệ thống máy PC chuẩn thường dùng 20 đường tín hiệu để truyền địa chỉ của các ô nhớ và của các thiết bị gắn với bus di chuyển dọc theo từng đường trong số 20 địa chỉ, máy tính PC chuẩn có thể định 2^{20} địa chỉ khác nhau (> 1 triệu địa chỉ)

Bus dữ liệu: Bus dữ liệu là các kênh truyền tải thông tin theo hai chiều giữa CPU và bộ nhớ hoặc các thiết bị ngoại vi vào ra. Bus dữ liệu được điều khiển bởi CPU để đọc hoặc viết các dữ liệu hoặc mã lệnh thực thi trong quá trình hoạt động của CPU. Độ rộng của bus dữ liệu nói chung sẽ xác định được lượng dữ liệu có thể truyền và trao đổi trên bus. Tốc độ truyền hay trao đổi dữ liệu thường được tính theo đơn vị là [byte/s]. Số lượng đường bit dữ liệu sẽ cho phép xác định được số lượng bit có thể lưu trữ trong mỗi khu vực tham chiếu trực tiếp. Nếu một bus dữ liệu có khả năng thực hiện một lần truyền trong $1\mu s$, thì bus dữ liệu 8bit sẽ có băng thông là 1Mbyte/s, bus 16bit sẽ có băng thông là 2Mbyte/s và bus 32bit sẽ có băng thông là 4Mbyte/s. Trong trường hợp bus dữ liệu 8bit với chu kỳ bus là $T=1\mu s$ (tức là sẽ truyền được 1byte/1chu kỳ) thì sẽ truyền được 1 Mbyte trong 1s hay 2Mbyte trong 2s.

Câu 16

- +Khả năng tự hoạt động ngay sau khi bật máy
- +Tác động đến máy tính khi chương trình kết thúc
- +Mức độ cần thiết cho sự hoạt động tối thiểu của máy tính
- +Khả năng điều khiển phần cứng
- +So với các phần mềm khác thì phần mềm HĐH được thiết kế và lập trình phức tạp hơn nhiều.
- +Phần mềm đầu tiên phải cài đặt vào hệ thống máy tính là phần mềm HĐH
- +Trên hệ thống máy tính có thể có nhiều HĐH nhưng tại một thời điểm chỉ có một HĐH hoạt động.
- +Là phần mềm bắt buộc phải có, trong khi các phần mềm khác có thể có hoặc không tùy vào nhu cầu của người sử dụng.

Câu 17:

Khái niệm HĐH :

- +là một hệ thống các chương trình để điều khiển máy tính. Hệ điều hành gắn liền với máy tính là “linh hồn” của máy tính.
- +Hệ điều hành là một hệ thống các chương trình chạy trên máy tính, dùng để điều hành quản lý các thiết bị phần cứng và các tài nguyên phần mềm trên máy tính.
- +HĐH đứng trung gian giữa người sử dụng và phần cứng máy tính, nó tạo ra một môi trường để người sử dụng có thể dùng máy tính tiện lợi, dễ dàng và hiệu quả hơn.
- +HĐH là 1 chương trình đặc biệt: Tự động thực hiện khi khởi động máy tính, và khi thoát tự động tắt máy. Phần nhiều (hầu hết) các chương trình khác hoạt động được là nhờ đến sự hỗ trợ của HĐH.

Câu 18:

Cấu trúc hệ điều hành

- +Hệ điều hành được tổ chức thành nhiều lớp : lớp dưới cùng là lớp vật lý, kế tiếp là lớp hạt nhân, lớp cuối cùng dùng để giao tiếp với người sử dụng
- Ở mức đơn giản, có thể nói cấu trúc hệ điều hành bao gồm bên ngoài là lớp vỏ (đại diện bởi chương trình Shell) và bên trong là lớp hạt nhân (kernel: đại diện bởi các hàm hạt nhân).
- Chương trình Shell là chương trình giao tiếp với người sử dụng thông qua các thiết bị nhập xuất. Còn các hàm hạt nhân là cơ sở để từ đó có thể xây dựng nên các chương trình còn lại.
- +Các thành phần chính của hệ điều hành : Hệ thống quản lý tiến trình (quản lý các chương trình đang chạy), hệ thống quản lý nhập xuất, hệ thống quản lý bộ nhớ (hệ thống lưu trữ) và hệ thống quản lý tập tin.

Câu 19:

*Quá trình khởi động : Kiểm tra hệ thống, khởi động các thiết bị cần thiết, khởi tạo môi trường, nạp chương trình Shell

*Quá trình phục vụ : Chương trình Shell liên tục kiểm tra và nhận tín hiệu từ các thiết bị nhập hoặc các ngõ nhập được qui ước và xuất ra các thiết bị xuất các trạng thái, thông tin, thông báo của hệ thống; hệ thống nhận và xử lý các tín hiệu hoặc yêu cầu từ các tiến trình.

*Quá trình kết thúc : Gởi tín hiệu yêu cầu đóng cho các tiến trình(thậm chí đóng cưỡng bức nếu sau khoảng thời gian T mà tiến trình chưa kết thúc), lưu các dữ liệu cần thiết vào các thiết bị bộ nhớ ngoài, đưa các thiết bị về trạng thái an toàn và cuối cùng là tắt máy.

Câu 20:

Lịch sử phát triển của hệ điều hành

-Thế hệ máy tính thứ I : Chương trình được viết trực tiếp bằng ngôn ngữ máy, chưa có một hệ điều hành thực sự.

-Thế hệ máy tính thứ II : Từ năm 1955-1965 hệ xử lý theo lô ra đời

-Thế hệ máy tính thứ III : Từ năm 1965-1980 các hệ xử lý đa chương được phát triển

-Thế hệ máy tính thứ IV : Từ năm 1980 các hệ thống máy tính mạng và phân tán phát triển.

Trong tương lai sẽ có nhiều hệ điều hành cực kỳ thông minh : Có thể giao tiếp được với con người bằng tiếng nói, có thể hiểu được con người qua phân tích cử chỉ, nét mặt, ánh mắt; thậm chí có thể lấy được suy nghĩ của con người qua sóng não(một loại sóng điện từ) hoặc các dây thần kinh.

Câu 21:

-Hệ thống máy tính phải có khả năng lưu lại được các thông tin, dữ liệu

-Hệ thống lưu trữ của các loại máy tính cao cấp thường có kích thước rất lớn nên sẽ chứa một lượng rất lớn dữ liệu.

-Lượng dữ liệu lưu trữ quá lớn thì nếu tổ chức không khéo thì việc quản lý hoặc truy cập sẽ gặp khó khăn và mất rất nhiều thời gian.

-Dữ liệu buộc phải lưu trữ trên bộ nhớ ngoài, mà tốc độ truy xuất trên bộ nhớ ngoài thì rất chậm(mỗi thao tác thường mất hơn 1ms), bắt buộc phải có các xử lý tối ưu tốc độ.

➔ Cần có các hình thức tổ chức, sắp xếp dữ liệu một cách hợp lý và xây dựng các thuật toán tối ưu để có thể truy cập nhanh chóng hiệu quả. Đây chính là việc xây dựng hệ điều hành có khả năng quản lý hệ thống tập tin tối ưu nhất.

Câu 22:

Tập tin : Khi lưu trữ nội dung thông tin ta cần gắn kèm những thuộc tính cần thiết như tên thông tin, kích thước, ngày/giờ tạo ra... để dễ dàng cho việc quản lý sau này. Tập hợp những thuộc tính đó được gọi là tập tin. Tập tin là đơn vị lưu trữ thông tin.

Thư mục : Khi có nhiều tập tin quá lớn thì cần phải tổ chức thành hệ thống thư mục, các tập tin có cùng tính chất hoặc cùng đặc điểm nào đó nên đưa vào cùng 1 thư mục để việc tìm kiếm và quản lý tập tin dễ dàng và nhanh chóng. Mỗi thư mục có thể có nhiều thư mục và tập tin bên trong

Câu 23:

Volume : Là một dãy các sector được tổ chức theo một kiến trúc hợp lý để có thể lưu trữ lên đó một hệ thống tập tin và thư mục.

Các thiết bị lưu trữ dung lượng lớn có thể tổ chức trên nhiều volume.

➔ Tất cả những tên gọi và khái niệm trên là do người viết hệ điều hành tạo ra. Máy tính chỉ biết xử lý, hoàn toàn không hiểu các khái niệm và tên gọi này.

Câu 24:

*Mô hình thuộc tính của tập tin

Xét 1 ví dụ sau : Các thuộc tính của tập tin tin nhắn của điện thoại di động có thể thiết kế như sau (giả sử kích thước của 1 sector là 262 bytes)

+Tên : Thuộc kiểu chuỗi có kích thước 11 ký tự - kết thúc bởi ký tự NULL, cho phép trùng, là tên người nhắn có lưu trong mục danh bạ hoặc số điện thoại nhắn tới.

+Nội dung tin nhắn : lưu tuần tự các ký tự (kết thúc bởi ký tự NULL, nếu chọn nhiều hơn 160Byte thì cắt ra nhiều đoạn 160Byte, 2 byte cuối của sector dùng để lưu chỉ số của sector kế tiếp), các ký tự trong nội dung được lưu theo bảng mã 1Byte hoặc bảng mã mở rộng 2Byte (bảng mã Unicode dựng sẵn).

+Bảng mã sử dụng : số nguyên 1Byte

+Chỉ số của dữ liệu hình ảnh đính kèm: lưu dạng số nguyên 1 Byte(bảng 255 nếu có, dữ liệu hình ảnh được lưu ở nơi khác).

+Trạng thái đã đọc hay chưa : số nguyên 1Byte, lưu số lần đã mở (bảng 0 nên chưa được xem, nếu quá 255 lần thì vẫn lưu 255).

+Ngày giờ gửi tới : Có thể lưu như ở ví dụ trước.

Câu 25:

*Mô hình thuộc tính của thư mục :

+Thiết kế tương tự như tập tin. Vì có khá nhiều thuộc tính giữa tập tin và thư mục giống nhau nên có thể tổ chức một mô hình cho cả 2. Lúc này thư mục được xem như là một tập tin đặc biệt và cần có 1 thuộc tính để phân biệt với tập tin bình thường.

Ví dụ : Nếu tập tin(bình thường) có 8 thuộc tính, thư mục có 6 thuộc tính và cả hai có 5 thuộc tính giống nhau thì tập tin tổng quát sẽ là $5+3+1+1=10$ thuộc tính.

*Mô hình chức năng

+Liệt kê danh sách tập tin : ở gốc/ thư mục con/cây thư mục – theo thứ tự của thuộc tính nào, các thuộc tính nào cần hiển thị và hiển thị như thế nào ...

+Đổi tên : Tên mới có bị trùng không, có hợp lệ không

+Đổi mật khẩu truy xuất tập tin : Yêu cầu nhập mật khẩu cũ (nếu có) và nhập đúng thì yêu cầu nhập mật khẩu mới 2 lần. Nếu hợp lệ thì mã hóa nội dung tập tin theo công thức tương ứng với mật khẩu mới.

+Xóa : xóa bình thường, xóa nhưng không mất, xóa mất hẳn, xóa cây thư mục, xóa rác, xóa phần mềm ...

+Đọc, ghi ...

Câu 26:

*Tổ chức hệ thống tập tin trên volume

1-Các nhận xét và phân tích cần thiết

-Phải xác định vị trí còn trống

-Mỗi sector (tổng quát hơn là mỗi block) chỉ thuộc tối đa 1 tập tin

-Tên và các thuộc tính của tập tin cần lưu riêng vào một vùng

-Phải có thông tin vị trí bắt đầu của nội dung tập tin

-Nội dung tập tin không phải bắt buộc liên tục

-Phải biết các vị trí chứa nội dung tập tin

-Phải biết các vị trí bị hư

-Nội dung tập tin nên lưu theo đơn vị là Cluster (là dãy N các sector liên tiếp để dễ quản lý và truy xuất được nhanh hơn).

Câu 27:

*Khái niệm :

-Đơn vị đọc ghi trên đĩa là sector, nhưng đơn vị lưu trữ nội dung tập tin không phải là một sector mà là một cluster gồm N sector liên tiếp ($N \geq 1$). Mỗi vị trí trong các phân tích trên sẽ là 1 cluster. Cluster chỉ tồn tại trên vùng dữ liệu(DATA)-nơi chứa nội dung tập tin

*Lý do phát sinh :

-Nếu sector trên vùng dữ liệu quá nhiều thì có thể sẽ khó hoặc không quản lý được, khi đó quản lý trên cluster sẽ dễ dàng hiệu quả hơn.

-Nếu nội dung tập tin thường chiếm nhiều sector và có thể không liên tục nên đặt đơn vị lưu trữ là cluster sẽ hạn chế được sự phân mảnh, đồng thời truy xuất một lần n sector liên tiếp nhanh hơn so với truy xuất N lần mà mỗi lần là 1 sector.

Câu 28:

*Hình thức tổ chức :

-Volume sẽ được chia thành 2 vùng : vùng dữ liệu (DATA) chứa nội dung tập tin và vùng hệ thống (SYSTEM) chứa các thông tin quản lý.

-Vùng SYSTEM có kích thước nhỏ hơn nhiều so với vùng DATA và phải truy xuất mỗi khi sử dụng Volume nên thường nằm ngay đầu Volume

-Trên vùng DATA là một dãy các Cluster liên tiếp được đánh chỉ số theo thứ tự tăng dần (bắt đầu từ 0, 1, 2 ... tùy theo hệ điều hành).

Ví dụ : Nếu Volume có kích thước 4014 sector, vùng SYSTEM chiếm 11 sector, mỗi cluster chiếm 4 sector, Cluster đầu tiên được đánh chỉ số là 2; thì phân bố cluster trên Volume sẽ như sau :

Câu 29:

*Kích thước Cluster

-Phụ thuộc nhiều yếu tố : Dung lượng Volume, tốc độ truy xuất một dãy sector, kích thước của đa số tập tin sẽ lưu vào, số cluster tối đa có thể quản lý, nhu cầu của người sử dụng, ...

-Kích thước Cluster càng lớn sẽ càng lãng phí đĩa, nhưng sẽ hạn chế sự phân mảnh tập tin và vì vậy có thể an toàn hơn và truy xuất nhanh hơn.

-Trên các đĩa cứng hiện tại thì sector thường có kích thước 512Bytes và Cluster thường chiếm 4, 8 hoặc 16 sector (và là lũy thừa của 2 đối với hệ điều hành Windows).

-Trên các đĩa/thẻ nhớ Flash thông dụng(dung lượng 128Mb đến 2Gbs), kích thước cluster nên là 8 hoặc 16Kb. Nếu có nhiều tập tin nhỏ thì nên nén/ghép lại thành 1 tập tin kích thước lớn hơn.

Câu 30:

*Khái niệm

-Được đưa ra để xác định danh sách các cluster chứa nội dung của một tập tin và các cluster trống

-Có thể dùng 1 bảng chung, cũng có thể tổ chức nhiều bảng. Thông thường mỗi phần tử trên bảng sẽ quản lý 1 cluster tương ứng trên vùng DATA.

*Lý do phát sinh

-Phải xác định các cluster còn trống để có thể chép tập tin vào Volume

-Phải xác định danh sách các cluster chứa nội dung của một tập tin để có thể đọc nội dung tập tin.

-Có thể lưu thông tin quản lý ngay trên Cluster nhưng khi đó quản lý và truy xuất rất chậm nên nhất thiết phải lập ra bảng này để nhanh hơn.

Câu 31:

*Hình thức tổ chức bảng quản lý cluster:

-Quản lý chuỗi các cluster chứa nội dung của 1 tập tin

- +Lưu trữ nội dung tập tin trên dãy cluster liên tiếp (danh sách đặc : mảng các phần tử)
- +Sử dụng cấu trúc danh sách liên kết (xâu)
- +Sử dụng cấu trúc xâu kết hợp với chỉ mục (index)
- +Sử dụng cấu trúc cây (kết hợp với chỉ mục)

-Quản lý chuỗi các cluster khi nội dung tập tin lưu trữ liên tiếp. Chỉ cần lưu vị trí cluster bắt đầu và kích thước tập tin (theo byte)

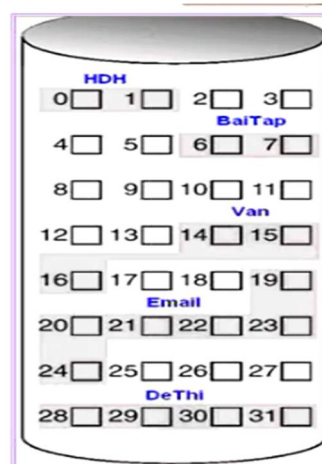
Ví dụ : Nếu cluster có kích thước là 1K và các tập tin trên Volume là : Hệ ĐIỀU

HÀNH(0,2006),

BaiTap(6,2007),Van(14,2050),Email

(19,6000),DeThi(28,4096) thì phân bố các

cluster của những tập tin đó sẽ như hình bên :



Câu 32:

-Quản lý chuỗi các cluster của tập tin bằng xâu : Mỗi cluster sẽ là 1 phần tử của xâu, gồm 2 vùng (vùng chứa nội dung tập tin và vùng liên kết chứa chỉ số cluster kế tiếp).

Ví dụ : Nếu tập tin chiếm các cluster là 1, 10, 22, 16 thì phân bố các cluster của những tập tin đó trong đĩa sẽ như hình kế bên :

Khuyết điểm : việc xác định danh sách các cluster của tập tin sẽ rất chậm – vì xem như phải truy xét luôn nội dung tập tin.

Câu 33:

Mỗi phần tử là 1 số nguyên dùng để quản lý 1 cluster (Phần tử K quản lý cluster K). Với qui định nếu phần tử K trên bảng có giá trị là L và L bằng :

+FREE/BAD : cluster K đang ở trạng thái trống/hư

+EOF : cluster K là cluster cuối của tập tin

+Khác : Cluster K chứa nội dung tập tin và kế sau là cluster L

Ví dụ 1: Nếu tập tin chiếm các cluster là 1, 10, 22, 16 thì nội dung bảng sẽ như sau :

	10									22					
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
eof						16									
16	17	18	19	20	21	22	23	24	25	26	27	28	...	...	

Ví dụ 2 : Với Volume X, nếu nội dung bảng quản lý cluster như sau:

		3	5	bad	eof	eof	11	free	7	bad	12	eof	free	...	free
0	1	2	3	4	5	6	7	8	9	10	11	12	13	...	1001

(các phần tử từ 13 đến 1001 đều có giá trị là FREE)

thì từ đây có thể xác định trên Volume đang có 2 cluster hư, 990 cluster trống, 8 cluster chứa nội dung tập tin. Cụ thể là :

- Các Cluster hư : số 4, số 10
- Các Cluster trống : 8,13,14,15,, 1001
- Các Cluster chứa nội dung tập tin : 2, 3, 5, 6, 7, 8,11, 12. Trong đó có 3 tập tin :
 - +Tập tin thứ 1 chiếm 3 Cluster theo đúng thứ tự là : 2, 3, 5
 - +Tập tin thứ 2 chiếm 1 Cluster duy nhất là : 6
 - +Tập tin thứ 3 chiếm 4 Cluster theo đúng thứ tự là : 9,7,11,12
- Quản lý chuỗi các Cluster của tập tin bằng xâu kết hợp với chỉ mục thông qua FAT : Hình thức này được áp dụng cho bảng FAT(File Allocation Table) của DOS và Windows, cũng như rất nhiều hệ điều hành trên các thiết bị kỹ thuật số. Có 3 loại bảng FAT : FAT12, FAT16 và FAT32 (mỗi phần tử của bảng có kích thước 12,16,32 bit). Nếu phần tử K có giá trị L thì trạng thái của Cluster K là :

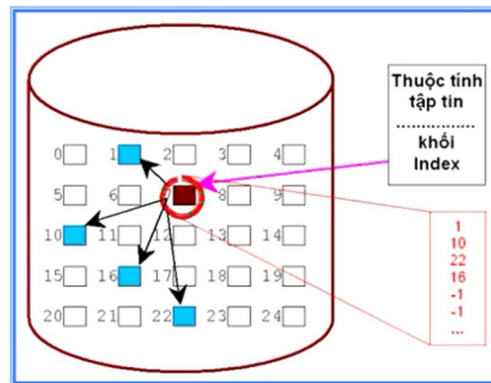
Trạng thái của Cluster K	Giá trị của L			Ghi chú
	FAT12	FAT16	FAT32	
Trống	0	0	0	= FREE
Hư	FF7	FFF7	0FFFFFF7	= BAD
Cluster cuối của tập tin	FFF	FFFF	0FFFFFFF	= EOF
Chứa nội dung tập tin – và có cluster kế sau là L	2..FEF	2..FFEF	2..0FFFFFFEF	

Chỉ áp dụng đối với ổ đĩa cứng có dung lượng lớn (từ vài Gigabyte trở lên)

Câu 34:

Khi số khối mà tập tin chiếm quá lớn thì quản lý bằng xâu(danh sách) sẽ dễ mất dữ liệu. Do đó, cần phải quản lý bằng cây nhiều nhánh, nút lá của cây sẽ chứa danh sách chỉ số trực tiếp của các Cluster

Ví dụ 1 : Nếu tập tin chiếm các khối là 1,10,22,16 thì nội dung khối chỉ mục (index) sẽ như hình vẽ bên cạnh



Câu 35:

Đối với hệ điều hành UNIX, người ta dùng kiểu kiến trúc I-NODE khác với kiến trúc FAT đối với hệ điều hành Windows. Thông tin được mô tả như sau:

- Mỗi khối có 4 KB
- Mỗi phần tử là 1 số nguyên 4 byte
- Trong mỗi bảng Indirect có 1024 con trỏ, mỗi con trỏ trỏ đến 1 bảng kế dưới.
- Trong mỗi bảng Single Indirect có 1024 chỉ số Cluster.
- Ở kiến trúc I-NODE gốc có 10 chỉ số Cluster → Khi đó, hệ thống kiến trúc I-Node dùng cây 1024 nhánh và mỗi tập tin có thể có hơn 1 tỷ khối.

Câu 36:

*Khái niệm:

-Bảng thư mục là một dãy phần tử (entry), mỗi entry chứa tên và các thuộc tính của tập tin

-Thường bảng thư mục gồm 2 loại : RDET(Root Directory Entry Table) và SDET(Sub Directory Entry Table).

Ví dụ : Nếu mô hình thuộc tính tập tin được thiết kế chỉ gồm các thành phần: Tên chính(là 1 chuỗi dài tối đa 4 ký tự), tên mở rộng (chuỗi tối đa 2 ký tự), kích thước tập tin (số nguyên 2 byte) thì RDET sẽ như sau :

	Tên chính				m.rộng		kthước		Tên chính				m.rộng		kthước				
off	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	...
	Entry 0								Entry 1								Entry 2, ..		

Câu 37:

*Khái niệm :

-Boot sector là nơi chứa đoạn chương trình nhỏ để nạp hệ điều hành khi khởi động máy(đoạn chương trình boot)

-Là nơi chứa các thông số quan trọng của Volume : loại volume, kích thước Volume, kích thước Cluster, kích thước bảng quản lý Cluster ...

-Cấu trúc Boot sector của kiến trúc FAT có dạng :

Offset	Số byte	Ý nghĩa
0	3	Lệnh nhảy đến đầu đoạn mã Boot
3	8	Tên công ty /version của HĐH
B	2	Số byte của sector (thường là 512)
D	1	Số sector của cluster (S_c)
E	2	Số sector trước bảng FAT (S_B)
10	1	Số lượng bảng FAT (N_F), thường là 2
11	2	Số Entry của RDET (S_R)
13	2	Số sector của volume (S_V),
16	2	Số sector của FAT (S_F)
20	4	Kích thước volume (nếu số 2 byte tại offset 13h là 0)
3E	1CF	Đoạn chương trình Boot nạp tiếp HĐH khi khởi động máy
1FE	2	Dấu hiệu kết thúc BootSector (luôn là AA55h)

Câu 38:

là bảng định vị File trên đĩa , bảng này liệt kê tuần tự số thứ tự của các cluster dành cho file lưu trữ trên đĩa. Cluster là một nhóm các sector liên kề nhau (còn gọi là liên cung). Số lượng sector có trong một Cluster là do hệ điều hành áp đặt cho từng loại đĩa có dung lượng thích hợp. Đĩa mềm thường được nhóm 2 sector thành một cluster. Với đĩa cứng, số sector trong một cluster có thể là 4 , 8,16, 32 ... Khi FAT đã chỉ định Cluster nào dành cho file thì toàn bộ các sector trong cluster đó bị file chiếm giữ kể cả khi trong thực tế file chỉ nằm trên một vài sector đầu của Cluster, còn các sector sau bỏ trống. Rõ ràng ta thấy số sector trong một cluster càng nhiều thì tình trạng lãng phí các sector bỏ trống mà file chiếm sẽ càng lớn.

Một số FAT: FAT12, FAT16 và FAT32

Câu 39:

Tiến trình là chương trình đang chạy. Mỗi tiến trình sở hữu một con trỏ lệnh, tập thanh ghi và các biến, và cần sử dụng một số tài nguyên như CPU, bộ nhớ chính, các tập tin, các thiết bị nhập xuất ...

Các máy tính chuẩn bình thường chỉ có 1 CPU và mỗi thời điểm chỉ thực hiện 1 lệnh của 1 chương trình, nhưng đại đa số hệ thống có nhu cầu chạy nhiều chương trình cùng lúc (hệ đa chương trình). Để đáp ứng hệ điều hành phải có 1 bộ điều phối (Scheduler) luân chuyển sự hoạt động của các tiến trình. Việc điều phối cần tạm dừng hoạt động của tiến trình đang xử lý để chuyển qua thực thi 1 tiến trình khác. Việc quản lý, điều phối tiến trình để có được hệ thống đa chương trên 1 CPU vẫn cần thiết ngay cả trên các máy nhiều CPU, vì khi này số tiến trình cần thực thi đồng hành thường vẫn nhiều hơn hẳn số CPU của máy.

Tiến trình chạy tới một lúc nào đó sẽ bị Scheduler bắt tạm dừng(trong thời gian nó dừng các tiến trình khác sẽ được chạy). Đến khi Scheduler cho nó được tiếp tục chạy lại, CPU sẽ được khôi phục lại giá trị bộ đếm chương trình(tức tiến trình được tiếp tục ở đúng vị trí bị ngưng trước đó), giá trị trong các thanh ghi cũng được phục hồi và các tài nguyên cần thiết tại thời điểm đó được cấp phát cho tiến trình.

Với cơ chế xin chia sẻ thời gian để đạt được hiệu ứng đa chương, hệ điều hành không chỉ giải quyết được một số nhu cầu bắt buộc của người dùng mà có thể làm cho hệ thống máy tính hoạt động với hiệu suất cao hơn.

Câu 40:

Mỗi tiến trình có thể ở một trong những trạng thái sau :

- Tạo mới : Tiến trình mới được tạo và được đưa vào hệ thống
- Trạng thái đang chạy (running) : CPU đang thực thi các chỉ thị của tiến trình.
- Trạng thái chờ (waiting) : còn gọi là treo (blocked), tiến trình đang đợi cấp phát tài nguyên hoặc 1 sự kiện và chưa sẵn sàng chạy.
- Trạng thái sẵn sàng(ready): Tiến trình đang chờ được cấp phát CPU để xử lý.
- Trạng thái kết thúc(terminated): Tiến trình hoàn tất

Mỗi thời điểm chỉ có 1 tiến trình ở trạng thái running, khi tiến trình không còn ở trạng thái running (bị Scheduler ngắt) thì 1 tiến trình trong trạng thái ready sẽ được chuyển sang trạng thái running).

Câu 41:

Ở các hệ điều hành đơn giản, mỗi tiến trình có 1 không gian địa chỉ và một dòng xử lý duy nhất. Tuy nhiên, hiện nay trên thực tế có nhiều lúc cần có nhiều dòng xử lý cùng chia sẻ một không gian địa chỉ - các dòng đó là tiểu trình. Nhiều hệ điều hành hiện đại cho phép 1 tiến trình tạo nhiều tiểu trình thực thi đồng thời, tiểu trình sử dụng các tài nguyên cục bộ của tiến trình (tài nguyên của tiến trình là tài nguyên toàn cục).

Các tiểu trình trong một tiến trình thì không hoàn toàn độc lập nhau như giữa các tiến trình. Các tiểu trình trong một tiến trình dùng chung không gian địa chỉ, nghĩa là các tiểu trình có thể dùng chung các biến toàn cục, tiểu trình có thể truy xuất đến bất kỳ địa chỉ nào trong không gian địa chỉ của tiến trình, tiểu trình có thể đọc/ghi lên các ngăn xếp của tiểu trình khác trong cùng 1 tiến trình.

Câu 42:

*HÀNG ĐỢI TIẾN TRÌNH

Khi các tiến trình được đưa vào hệ thống, chúng được đặt vào hàng đợi. Các tiến trình đang sẵn sàng thực thi được giữ trên hàng đợi sẵn sàng (ready). Hàng đợi này thường được lưu như một danh sách liên kết, đầu hàng đợi sử dụng 2 con trỏ; một chỉ đến đầu danh sách và 1 chỉ vào cuối danh sách, trong mỗi khối điều khiển tiến trình có một con trỏ chỉ đến khối điều khiển tiến trình kế tiếp. Khi một tiến trình ở trạng thái running bị ngắt vì phải đợi hoàn thành một yêu cầu nhập xuất trên 1 thiết bị nào đó thì nó cũng được đưa vào hàng đợi thiết bị tương ứng.

Câu 43:

- Tận dụng CPU tối đa: cần trên 40% cho hệ thống tải nhẹ, và tới 90% cho hệ thống tải nặng
- Thông lượng tối đa : số lượng tiến trình được hoàn thành trên một đơn vị thời gian gọi là thông lượng(throughput), và dĩ nhiên số lượng này càng nhiều thì càng tốt.
- Thời gian hoàn thành : Khoảng thời gian mà tiến trình phải lưu lại trong hệ thống
- Thời gian chờ : Là tổng thời gian chờ trong hàng đợi
- Thời gian đáp ứng : Thời gian hoàn thành có khi không phải là tiêu chuẩn tốt nhất, mà thời gian từ lúc gửi yêu cầu cho tới khi đáp ứng đầu tiên được tạo ra (thời gian đáp ứng – response time) lại đáng quan tâm hơn.

Câu 44:

Là giải thuật điều phối biểu CPU đơn giản nhất: đến trước, được phục vụ trước (first come, first-served). Việc cài đặt chính sách FCFS được quản lý dễ dàng với hàng đợi FIFO. Khi một tiến trình đi vào hàng đợi sẵn sàng, khối điều khiển tiến trình (PCB) của nó được nối vào cuối hàng đợi. Khi tiến trình running bị ngắt, tiến trình ở đầu hàng đợi ở trạng thái ready sẽ được chọn sang trạng thái running và được lấy ra khỏi hàng đợi.

Thời gian chờ trung bình trong chiến lược này có thể sẽ khá dài.

Ví dụ :

Tiến trình	Thời điểm vào	Thời gian xử lý
P1	0	24
P2	1	3
P3	I 2	3

Thì thứ tự cấp phát CPU sẽ là:

P1	P2	P3
0	24	27

- Thời gian chờ của tiến trình P1 là 0
- Thời gian chờ của tiến trình P2 là $(24-1)$ I
- Thời gian chờ của tiến trình P3 là $(24+3-2)$
- Thời gian chờ trung bình: $(0+23+25)/3 = 16$

Câu 45:

Sử dụng điều phối không trung dụng CPU. Một khi CPU được cấp phát tới một quá trình, tiến trình đó sẽ giữ CPU cho tới khi nó giải phóng CPU bằng cách kết thúc hay yêu cầu nhập/xuất. Giải thuật FCFS đặc biệt không phù hợp đối với hệ thống chia sẻ thời gian. Một số sách gọi giải thuật FCFS là phương pháp điều phối tiến trình độc quyền.

Câu 46:

Danh sách sẵn sàng được xử lý như một danh sách vòng, bộ điều phối lần lượt cấp phát cho từng tiến trình trong danh sách một khoảng thời gian sử dụng CPU gọi là *quantum*. Đây là một giải thuật điều phối không độc quyền : khi một tiến trình sử dụng CPU đến hết thời gian quantum dành cho nó, hệ điều hành thu hồi CPU và cấp cho tiến trình kế tiếp trong danh sách. Nếu tiến trình bị khóa hay kết thúc trước khi sử dụng hết thời gian quantum, hệ điều hành cũng lập tức cấp phát CPU cho tiến trình khác. Khi tiến trình tiêu thụ hết thời gian CPU dành cho nó mà chưa hoàn tất, tiến trình được đưa trở lại vào cuối danh sách sẵn sàng để đợi được cấp CPU trong lượt kế tiếp.

Câu 47:

Đây là một trường hợp đặc biệt của giải thuật điều phối với độ ưu tiên. Trong giải thuật này, độ ưu tiên p được gán cho mỗi tiến trình là nghịch đảo của thời gian xử lý t mà tiến trình yêu cầu : $p = 1/t$. Khi CPU được tự do, nó sẽ được cấp phát cho tiến trình yêu cầu ít thời gian nhất để kết thúc- tiến trình ngắn nhất. Giải thuật này cũng có thể độc quyền hay không độc quyền. Sự chọn lựa xảy ra khi có một tiến trình mới được đưa vào danh sách sẵn sàng trong khi một tiến trình khác đang xử lý. Tiến trình mới có thể sở hữu một yêu cầu thời gian sử dụng CPU cho lần tiếp theo (CPU-burst) ngắn hơn thời gian còn lại mà tiến trình hiện hành cần xử lý. Giải thuật SJF không độc quyền sẽ dừng hoạt động của tiến trình hiện hành, trong khi giải thuật độc quyền sẽ cho phép tiến trình hiện hành tiếp tục xử lý.

Câu 48:

Điều phối không trung dụng CPU(không độc quyền) và điều phối trung dụng CPU(độc quyền CPU)

Thuật toán điều phối cần xem xét và quyết định thời điểm chuyển đổi CPU giữa các tiến trình. Hệ điều hành có thể thực hiện cơ chế điều phối theo nguyên lý *độc quyền* hoặc *không độc quyền*.

Điều phối có trung dụng CPU(độc quyền) : Nguyên lý điều phối *độc quyền* cho phép một tiến trình khi nhận được CPU sẽ có quyền độc chiếm CPU đến khi hoàn tất xử lý hoặc tự nguyện giải phóng CPU. Khi đó quyết định điều phối CPU sẽ xảy ra trong các tình huống sau:

Khi tiến trình chuyển từ trạng thái đang xử lý(running) sang trạng thái bị khóa blocked (ví dụ chờ một thao tác nhập xuất hay chờ một tiến trình con kết thúc...).

Khi tiến trình kết thúc.

Các giải thuật độc quyền thường đơn giản và dễ cài đặt. Tuy nhiên chúng thường không thích hợp với các hệ thống tổng quát nhiều người dùng, vì nếu cho phép một tiến trình có quyền xử lý bao lâu tùy ý, có nghĩa là tiến trình này có thể giữ CPU một thời gian không xác định, có thể ngăn cản những tiến trình còn lại trong hệ thống có một cơ hội để xử lý.

Điều phối không trung dụng CPU(không độc quyền) : Ngược với nguyên lý độc quyền, điều phối theo nguyên lý *không độc quyền* cho phép tạm dừng hoạt động của một tiến trình đang sẵn sàng xử lý. Khi một tiến trình nhận được CPU, nó vẫn được sử dụng CPU đến khi hoàn tất hoặc tự nguyện giải phóng CPU, nhưng một tiến trình khác có độ ưu tiên có thể dành quyền sử dụng CPU của tiến trình ban đầu. Như vậy là tiến trình có thể bị tạm dừng hoạt động bất cứ lúc nào mà không được báo trước, để tiến trình khác xử lý. Các quyết định điều phối xảy ra khi :

Khi tiến trình chuyển từ trạng thái đang xử lý (running) sang trạng thái bị khóa blocked (ví dụ chờ một thao tác nhập xuất hay chờ một tiến trình con kết thúc...).

Khi tiến trình chuyển từ trạng thái đang xử lý (running) sang trạng thái ready (ví dụ xảy ra một ngắt).

Khi tiến trình chuyển từ trạng thái chờ (blocked) sang trạng thái ready (ví dụ một thao tác nhập/xuất hoàn tất).

Khi tiến trình kết thúc.

Các thuật toán điều phối theo nguyên tắc không độc quyền ngăn cản được tình trạng một tiến trình độc chiếm CPU, nhưng việc tạm dừng một tiến trình có thể dẫn đến các mâu thuẫn trong truy xuất, đòi hỏi phải sử dụng một phương pháp đồng bộ hóa thích hợp để giải quyết.

Trong các hệ thống sử dụng nguyên lý điều phối độc quyền có thể xảy ra tình trạng các tác vụ cần thời gian xử lý ngắn phải chờ tác vụ xử lý với thời gian rất dài hoàn tất! Nguyên lý điều phối độc quyền thường chỉ thích hợp với các hệ xử lý theo lô.

Đối với các hệ thống tương tác(time sharing), các hệ thời gian thực (real time), cần phải sử dụng nguyên lý điều phối không độc quyền để các tiến trình quan trọng có cơ hội hồi đáp kịp thời. Tuy nhiên thực hiện điều phối theo nguyên lý không độc quyền đòi hỏi những cơ chế phức tạp trong việc phân định độ ưu tiên, và phát sinh thêm chi phí khi chuyển đổi CPU qua lại giữa các tiến trình.

Câu 49: Đặc điểm của hệ phân tán

Hệ phân tán có các đặc điểm cơ bản là *Tính chia xẻ tài nguyên, Tính mở, Khả năng song song, Tính mở rộng, Khả năng thứ lỗi, Tính trong suốt.*

Tính chia xẻ tài nguyên

Trong hệ phân tán, chia xẻ tài nguyên được hiểu là tài nguyên của hệ thống được các QT chia xẻ (sử dụng chung) mà không bị hạn chế bởi tình trạng phân tán tài nguyên theo vị trí địa lý.

Việc chia xẻ tài nguyên trên hệ phân tán - trong đó tài nguyên bị lệ thuộc về mặt vật lý với một máy tính nào đó - được thực hiện thông qua truyền thông. Để chia xẻ tài nguyên một cách hiệu quả thì *mỗi tài nguyên cần phải được quản lý* bởi một chương trình có giao diện truyền thông, *các tài nguyên có thể truy nhập, cập nhật được* một cách tin cậy và nhất quán. Quản lý tài nguyên ở đây bao gồm lập kế hoạch và dự phòng, đặt tên các lớp tài nguyên, cho phép tài nguyên được truy cập từ nơi khác, ánh xạ tên tài nguyên vào địa chỉ truyền thông ...

Tính mở

Tính mở của hệ phân tán được thể hiện là hệ thống có thể *được tạo nên từ nhiều loại phần cứng và phần mềm* của nhiều nhà cung cấp khác nhau với điều kiện các thành phần này phải *theo một tiêu*

chuẩn chung (liên quan đến HĐH là *tính đa dạng tài nguyên*; liên quan đến nhà cung cấp tài nguyên là *tính chuẩn*). Vai trò của ASP và SPI trong HĐH đã được trình bày trong chương 1.

Tính mở của Hệ phân tán được xem xét theo mức độ *bổ sung thêm các dịch vụ chia sẻ tài nguyên mà không phá hỏng hay nhân đôi các dịch vụ đang tồn tại*. Tính mở được hoàn thiện bằng cách xác định hay phân định rõ các giao diện chính của hệ phân tán và làm cho nó tương thích với các nhà phát triển phần mềm (tức là các giao diện chính của HĐH phân tán cần phổ dụng).

Tính mở của HĐH phân tán được thi hành dựa trên việc cung cấp *cơ chế truyền thông giữa các QT và công khai các giao diện được dùng để truy cập tài nguyên chung*.

Khả năng song song

Hệ phân tán hoạt động trên một mạng truyền thông có nhiều máy tính, mỗi máy tính có thể có một hoặc nhiều CPU. Trong cùng một thời điểm nếu có từ hai QT trở lên cùng tồn tại, ta nói rằng chúng được thực hiện đồng thời. Việc thực hiện các QT đồng thời theo cơ chế phân chia thời gian (một CPU) hay song song (nhiều CPU).

Khả năng làm việc song song trong hệ phân tán được thi hành do hai tình huống:

- Nhiều người sử dụng đồng thời đưa ra các lệnh hay tương tác với chương trình ứng dụng (đồng thời xuất hiện nhiều QT khách).

- Nhiều QT phục vụ chạy đồng thời, mỗi QT đáp ứng yêu cầu của một trong số các QT Khách.

Từ điều kiện đa xử lý, khả năng song song của hệ thống phân tán trở thành một thuộc tính của nó.

Khả năng mở rộng

Hệ phân tán có khả năng hoạt động tốt và hiệu quả ở nhiều mức khác nhau. Một hệ phân tán nhỏ nhất có thể hoạt động chỉ cần hai trạm làm việc và một phục vụ file. Các hệ lớn có thể bao gồm hàng nghìn máy tính, nhiều phục vụ File và phục vụ máy in ...

Khả năng mở rộng của một hệ phân tán được đặc trưng bởi *tính không thay đổi phần mềm hệ thống và phần mềm ứng dụng khi hệ thống được mở rộng*.

Khả năng thứ lỗi

Hệ phân tán cung cấp khả năng sẵn sàng cao để đối phó với các sai hỏng phần cứng. Khả năng sẵn sàng của hệ thống được đo bằng tỷ lệ thời gian mà hệ thống sẵn sàng làm việc so với thời gian có sự cố. Khi một máy trên mạng sai hỏng thì chỉ có công việc liên quan đến các thành phần sai hỏng bị ảnh hưởng. Người sử dụng có thể chuyển đến một trạm khác nếu máy họ đang sử dụng bị hỏng, một QT phục vụ có thể được khởi động lại trên một máy khác.

Tính trong suốt

Tính trong suốt của hệ phân tán được hiểu như là *sự che khuất đi các thành phần riêng biệt của hệ thống máy tính (phần cứng và phần mềm) đối với người sử*

Câu 50:

Tính hiệu quả

Tính hiệu quả trở nên phức tạp hơn so với HĐH tập trung do phải tính đến chi phí phải trả cho bài toán truyền thông mà trước đây trong HĐH tập trung đã bỏ qua yếu tố này. Truyền thông CTĐ trong môi trường phân tán địa lý dẫn đến độ trễ tới hàng micro giây, mili giây thậm chí là hàng giây và tạo ra một yếu tố phức tạp trong việc đánh giá mức độ hiệu quả của hệ thống.

Nguồn gốc của "độ trễ" là do bổ sung nhiều yếu tố mới vào HĐH phân tán so với HĐH tập trung, đó là độ trễ do nhân bản dữ liệu, độ trễ do tính toán đến tổng phí theo các giao thức truyền thông ở các mức độ khác nhau và sự phân tán tải của hệ thống.

Tính mềm dẻo

Theo cách nhìn của người sử dụng, tính mềm dẻo được thể hiện thông qua tính thân thiện của hệ thống, tính tự do của người dùng khi sử dụng hệ thống. Tính thân thiện được hiểu rất rộng như dễ dàng sử dụng giao diện hệ thống, khả năng ánh xạ quá trình tính toán trong không gian bài toán tới hệ thống. Tiếp cận hướng đối tượng là chiến lược phổ biến để hoàn thành mục tiêu này. Tính thân thiện cũng liên kết với các tính chất nhất quán và tính tin cậy. Các hệ thống nhất quán và đáng tin cậy không có những hạn chế vô lý. Nó cần cung cấp môi trường hoạt động thích hợp trong đó các tool và dịch vụ mới dễ dàng được xây dựng.

Tính nhất quán

Tính nhất quán trở nên khó khăn hơn khi thi hành trong hệ phân tán: thiếu vắng thông tin toàn cục, tiềm tàng nhân bản và phân hoạch dữ liệu mạnh, khả năng xảy ra lỗi thành phần, mối liên quan phức tạp các mô đun thành phần; tất cả các điều đó đều tham gia vào sự thiếu nhất quán của hệ thống. Theo phương diện người dùng, một hệ thống là nhất quán nếu như có được tính đồng nhất khi sử dụng và ứng xử hệ thống có thể khẳng định trước. Hơn nữa, hệ thống phải đủ năng lực duy trì tình trạng toàn vẹn nhờ cơ chế điều khiển đồng thời chính xác và các thủ tục kiểm soát lỗi và khôi phục. Điều khiển nhất quán trong dữ liệu và file (hoặc CSDL trong hệ thống định hướng giao dịch) là những vấn đề còn được bàn luận trong hệ thống file phân tán.

Tính mạnh mẽ

Bài toán tính mạnh mẽ càng trở nên quan trọng hơn trong hệ thống phân tán: lỗi kết nối truyền thông, lỗi tại nút xử lý và lỗi trong các hệ thống Client/Server là thường xuyên hơn so với hệ thống máy tính tập trung.

Câu 51

Trạng thái deadlock: Một tập hợp các tiến trình được định nghĩa ở trong tình trạng *tắc nghẽn* khi mỗi tiến trình trong tập hợp đều chờ đợi một sự kiện mà chỉ có một tiến trình khác trong tập hợp mới có thể phát sinh được.

Nói cách khác, mỗi tiến trình trong tập hợp đều chờ được cấp phát một tài nguyên hiện đang bị một tiến trình khác cũng ở trạng thái blocked chiếm giữ. Như vậy không có tiến trình nào có thể tiếp tục xử lý, cũng như giải phóng tài nguyên cho tiến trình khác sử dụng, tất cả các tiến trình trong tập hợp đều bị khóa vĩnh viễn !

Câu 52:

Cơ chế ngăn deadlock: Khi đã phát hiện được tắc nghẽn, có hai lựa chọn chính để hiệu chỉnh tắc nghẽn :

Đình chỉ hoạt động của các tiến trình liên quan. Cách tiếp cận này dựa trên việc thu hồi lại các tài nguyên của những tiến trình bị kết thúc. Có thể sử dụng một trong hai phương pháp sau :

-Đình chỉ tất cả các tiến trình trong tình trạng tắc nghẽn

-Đình chỉ từng tiến trình liên quan cho đến khi không còn chu trình gây tắc nghẽn : để chọn được tiến trình thích hợp bị đình chỉ, phải dựa vào các yếu tố như độ ưu tiên, thời gian đã xử lý, số lượng tài nguyên đang chiếm giữ, số lượng tài nguyên yêu cầu...

Thu hồi tài nguyên

Có thể hiệu chỉnh tắc nghẽn bằng cách thu hồi một số tài nguyên từ các tiến trình và cấp phát các tài nguyên này cho những tiến trình khác cho đến khi loại bỏ được chu trình tắc nghẽn. Cần giải quyết 3 vấn đề sau:

- Chọn lựa một nạn nhân: tiến trình nào sẽ bị thu hồi tài nguyên ? và thu hồi những tài nguyên nào ?
- Trở lại trạng thái trước tắc nghẽn: khi thu hồi tài nguyên của một tiến trình, cần phải phục hồi trạng thái của tiến trình trở lại trạng thái gần nhất trước đó mà không xảy ra tắc nghẽn.
- Tình trạng « đói tài nguyên »: làm sao bảo đảm rằng không có một tiến trình luôn luôn bị thu hồi tài nguyên ?

Câu 53:

Cơ chế phân trang: 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.

Câu 54:

Cơ chế phân đoạn:

Trong nhiều trường 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 giá trị 0. Chiều dài của các segment có thể khác nhau. Tuy nhiên chiều dài của segment có thể thay đổi trong lúc chạy. Vì mỗi 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 chỉ số segment, xác định segment được truy xuất, và phần còn lại là địa chỉ dùng để xác định vị trí bộ nhớ trong segment đó.

Câu 55:

Khi tiến trình được đưa vào hệ thống, một vùng nhớ trống vừa đúng kích thước của nó sẽ được cấp phát, 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.

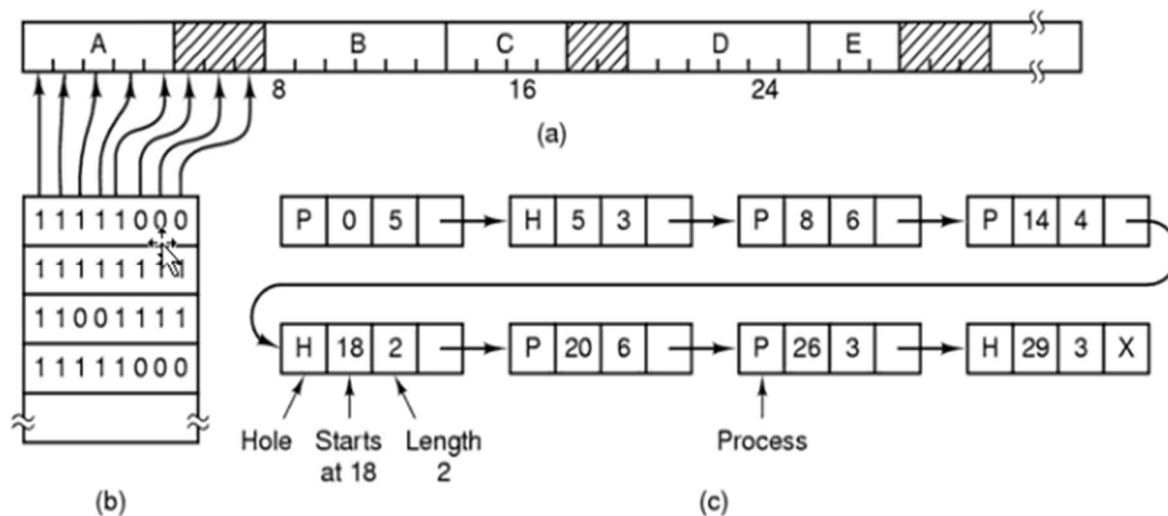
Cơ chế quản lý vùng nhớ trống:

-Dùng cơ chế 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 nhớ còn trống, bit 1 nghĩa là vùng nhớ đã được cấp phát.

Câu 56:

-Dùng cơ chế liên kết(link): Mỗi phần tử của danh sách liên kết quản lý một vùng nhớ, mỗi phần tử có các thông tin như: trạng thái vùng(H: trống, P: tiến trình/không trống), vị trí bắt đầu, kích thước, vị trí vùng kế tiếp.

Sơ đồ minh họa:



Câu 57:

Khái niệm swapping

Khi máy tính cần chạy những chương trình lớn hơn khả năng có thể của bộ nhớ vật lý, hệ điều hành sẽ sử dụng thuật toán swapping. Swapping sẽ dùng đến những mảng bộ nhớ tạm được lưu trên đĩa cứng, trong khi phần dữ liệu khác vẫn được chuyển vào RAM để tăng hiệu năng sử dụng.

Ứng dụng với hệ điều hành Linux

Đối với hệ điều hành Linux: Bộ nhớ vật lý được chia thành các pages, swapping là một tiến trình thực hiện việc copy một page của bộ nhớ đến một không gian đã được cấu hình trước trên đĩa cứng (gọi

là swap space) để giải phóng các pages của bộ nhớ. Tổng dung lượng của RAM và swap space chính là tổng số bộ nhớ ảo (virtual memory).

Có 2 loại swap space. Đó là: swap partition và swap file.

+ Swap partition là một phân vùng độc lập nằm trên đĩa cứng, chỉ có mục đích là hoán đổi (swapping), không có file nào khác nằm trên đó.

+ Swap file là một file riêng biệt nằm trên hệ thống, nó có thể nằm giữa hệ thống và các file dữ liệu.

+ Một quá trình có thể được hoán đổi tạm thời ra khỏi bộ nhớ cho nơi lưu trữ (backing store) và sau đó đưa trở lại vào bộ nhớ để tiếp tục thực hiện.

Câu 58: Sử dụng bộ nhớ hệ điều hành

+ Bộ xử lý trung tâm (CPU) là thiết bị xử lý chung cho các tác vụ của một máy tính (xem thêm CPU), trong quá trình làm việc của CPU luôn cần xử lý với dữ liệu, nói một cách đơn giản hơn thì CPU làm việc với các dữ liệu và lệnh đưa vào để biến chúng thành kết quả một cách trực tiếp hoặc thông qua các thiết bị khác hỗ trợ cùng xử lý với nó.

+ Trong quá trình xử lý với dữ liệu, quy trình sử dụng dữ liệu bắt đầu từ chính bộ nhớ đệm (cache) gần nó nhất (cache có thể nằm trong CPU ngày nay, nằm gần nó với các CPU gắn trên thẻ cắm một hàng slot 1 hoặc trên bo mạch chủ trong các thời gian trước đây). Nếu cache không có dữ liệu cần thiết, CPU sẽ tìm đến bộ nhớ chính (RAM). Nếu vẫn chưa thấy, chúng tìm trên các bộ nhớ chậm hơn như trên ổ đĩa cứng và các thiết bị nhớ còn lại.

+ CPU làm việc hoàn toàn theo sự quản lý của hệ điều hành, do đó việc tìm kiếm và xử lý trên phụ thuộc vào hệ điều hành. Hệ điều hành có thể quy hoạch bộ nhớ cho hệ thống sao cho mọi quá trình

làm việc của CPU là tối ưu nhất cho đến thời điểm ra đời của hệ điều hành và năng lực, tập lệnh của CPU.

Câu 59: Quản lý bộ nhớ trong linux

+Trong hệ thống máy tính, bộ nhớ là một tài nguyên hữu hạn. Cho dù có bao nhiêu bộ nhớ đi chăng nữa thì vẫn không đáp ứng đủ nhu cầu của người sử dụng. Các máy tính cá nhân hiện nay đã trang bị ít nhất 1GB bộ nhớ. Các máy chủ server có thể lên đến hàng GB bộ nhớ. Thế nhưng nhu cầu bộ nhớ vẫn không được thỏa mãn.

+Linux có cách tiếp cận và quản lý bộ nhớ rất rõ ràng. Các ứng dụng trên linux không bao giờ được phép truy cập trực tiếp vào địa chỉ vật lý của bộ nhớ. Linux cung cấp cho các chương trình chạy dưới HĐH- còn gọi là tiến trình – một mô hình đánh địa chỉ phẳng không phân đoạn segment : offset như DOS. Mỗi tiến trình chỉ thấy được một vùng không gian địa chỉ của riêng nó. Hầu như tất cả các phiên bản của UNIX đều cung cấp cách bảo vệ bộ nhớ theo cơ chế bảo đảm không có tiến trình nào có thể ghi đè lên vùng nhớ của tiến trình khác đang hoạt động hoặc vùng nhớ của hệ thống. Nói chung, bộ nhớ mà hệ thống cấp phát cho một tiến trình không thể nào đọc hoặc ghi bởi một tiến trình khác.

+Trong hầu hết các hệ thống Linux và Unix, con trỏ được sử dụng là một số 32 bit trỏ đến một ô nhớ cụ thể. Với 32 bit, hệ thống có thể đánh địa chỉ lên đến 4GB bộ nhớ. Mô hình bộ nhớ phẳng này dễ truy xuất và xử lý hơn bộ nhớ phân đoạn segment : offset. Ngoài ra, một vài hệ thống còn sử dụng mô hình địa chỉ 64 bit, như vậy, không gian địa chỉ có thể mở rộng ra đến terabyte.

+Để tăng dung lượng bộ nhớ sẵn có, Linux còn cài đặt chương trình phân trang đĩa tức là một lượng không gian hoán đổi nào đó có thể phân bố trên đĩa. Khi hệ thống yêu cầu nhiều bộ nhớ vật lý, nó sẽ đưa các trang không hoạt động ra đĩa, nhờ vậy bạn có thể chạy những ứng dụng lớn hơn và cùng lúc hỗ trợ nhiều người sử dụng. Tuy vậy, việc hoán đổi không thay được RAM vật lý, nó chậm hơn vì cần nhiều thời gian để truy nhập đĩa. Kernel cũng cài đặt khối bộ nhớ hợp nhất cho các chương trình người sử dụng và bộ nhớ đệm đĩa tạm thời (diskcache). Theo cách này, tất cả bộ nhớ trống dành để nhớ tạm và bộ nhớ đệm (cache) sẽ giảm xuống khi bộ xử lý chạy những chương trình lớn.

Câu 60: Các dịch vụ chính của hệ điều hành

Hệ điều hành cung cấp một môi trường để thi hành các chương trình, bằng cách cung cấp các dịch vụ cho chương trình và cho người sử dụng. Các dịch vụ này trên mỗi hệ thống là khác nhau nhưng cũng có những lớp chung. Các dịch vụ này giúp cho các lập trình viên thuận tiện hơn và việc lập trình dễ dàng hơn.

-*Thi hành chương trình*: hệ thống phải có khả năng nạp chương trình vào bộ nhớ và thi hành nó. Chương trình phải chấm dứt thi hành theo cách thông thường hay bất thường (có lỗi).

-*Thao tác nhập xuất*: Một chương trình thi hành có thể yêu cầu nhập xuất. Nhập xuất này có thể là tập tin hay thiết bị. Đối với thiết bị có một hàm đặc biệt được thi hành. Để tăng hiệu quả, người sử dụng không truy xuất trực tiếp các thiết bị nhập xuất mà thông qua cách thức do hệ điều hành cung cấp.

-*Thao tác trên hệ thống tập tin*.

-*Thông tin*: có nhiều tình huống một tiến trình cần trao đổi thông tin với một tiến trình khác. Có hai cách thực hiện: Một là thực hiện thay thế tiến trình trên cùng máy tính, hai là thay thế tiến trình trên hệ thống khác trong hệ thống mạng. Thông tin có thể được cài đặt qua chia sẻ bộ nhớ, hoặc bằng kỹ thuật chuyển thông điệp. Việc chuyển thông tin được thực hiện bởi hệ điều hành.

-*Phát hiện lỗi*: hệ điều hành phải có khả năng báo lỗi. Lỗi xảy ra có thể do CPU, bộ nhớ, trong thiết bị nhập xuất, ... hay trong các chương trình. Đối với mỗi dạng lỗi, hệ điều hành sẽ có cách giải quyết tương ứng.