

(L.O.1.1) Chương trình nào sau đây không được xem là một chương trình hệ thống (system program)?

- A. Loader
- B. Chương trình tìm lỗi (debugger)
- C. Media player
- D. Tạo, xóa, copy file.

(L.O.1.1) Những module nào sau đây thuộc về kernel của hệ điều hành?

- A. Giao diện đồ họa người dùng (GUI)
- B. Quản lý bộ nhớ
- C. Chương trình hệ thống
- D. Ngôn ngữ biên dịch

(L.O.1.1) Các dịch vụ của hệ điều hành nhằm cung cấp các chức năng nào sau đây hướng quản lý hệ thống một cách hiệu quả:

- A. Thực thi chương trình, Xử lý hệ thống tập tin, Phát hiện lỗi.
- B. Cấp phát tài nguyên, Quản lý tài khoản và Bảo mật, bảo vệ.
- C. Giao tiếp quá trình, Giao diện làm việc với người sử dụng, và Xử lý hệ thống tập tin.
- D. Tất cả đều sai.

(L.O.2.1) Nếu như tất cả quá trình đều thuộc kiểu I/O bound, hàng đợi ready sẽ gần như luôn _____ và bộ định thời gian hạn sẽ có _____ việc để làm.

- A. đầy, ít
- B. đầy, nhiều
- C. trống, ít
- D. trống, nhiều

(L.O.1.2) Trong giao tiếp giữa các quá trình bằng chia sẻ bộ nhớ, vùng không gian chia sẻ được tạo ra ở đâu?

- A. User space
 - B. Kernel space
 - C. Có thể ở user space hoặc kernel space
 - D. Phụ thuộc vào cách hiện thực
- Cho chương trình sau:

```
main()
{
    if(fork(>0)
    sleep(100);
}
```

Khi thực thi chương trình trên, sẽ tạo ra:

- A. Một quá trình orphan
- B. Một quá trình zombie
- C. Một quá trình thực thi vô hạn định
- D. Không câu nào đúng

Một quá trình tương tác multithreading sẽ tăng tính đáp ứng cho người dùng bởi vì:

- A. Tiếp tục thực thi ngay cả khi có 1 thread đang bị block.
- B. Các thread được thực thi đồng thời.
- C. Yêu cầu người dùng quyết định thứ tự thực thi của các thread.
- D. Không câu nào đúng.

(L.O.1.1) Chọn câu đúng về so sánh giữa hai hệ thống multiprogramming và time-sharing:

- A. Multiprogramming làm giảm hiệu suất sử dụng (CPU utilization) còn time-sharing thì làm tăng hiệu suất sử dụng CPU.
- B. Multiprogramming giữ nhiều công việc trong bộ nhớ còn time-sharing chỉ giữ một.
- C. Hệ thống Multiprogramming không cung cấp khả năng tương tác hiệu quả với người dùng, còn time-sharing cung cấp khả năng tương tác hiệu quả với người dùng.
- D. Cả A, B, và C đều sai.

(L.O.1.1) Chọn phát biểu đúng:

- A. Giao diện giữa quá trình và hệ điều hành là cửa sổ console
- B. Giao diện giữa quá trình và hệ điều hành là màn hình desktop
- C. Giao diện giữa quá trình và kernel hệ điều hành là các system call
- D. Giao diện giữa quá trình và hệ điều hành là tất cả những giao diện trên

(L.O.2.1) Trong mô hình 5 trạng thái của quá trình, các trạng thái mà một quá trình thực thi thành công bắt buộc phải trải qua là:

- A. new - ready - running - terminated
- B. new - ready - waiting - terminated
- C. ready - terminated
- D. new - ready - waiting - ready - running

(L.O.2.1) Giải thuật định thời CPU thích hợp cho hệ thống cần có giao tiếp với người dùng là giải thuật

- A. SJF (Shortest Job First)
- B. RR (Round Robin)
- C. FCFS (First Come First Served)
- D. Priority

(L.O.1.2) Để đồng bộ các thread trong cùng một process, ta có thể dùng:

- A. Sử dụng semaphore.
- B. Sử dụng monitor.
- C. Sử dụng giải thuật bakery.
- D. Cả 3 câu (A), (B), (C) đúng.

(L.O.1.2) Khởi tạo một mảng mutex nhị phân $m[0] \dots m[4]$ và 5 quá trình $P[0] \dots P[4]$.

Cho đoạn chương trình của $P[i]$ như sau:

```
wait (m[i]); wait(m[(i+1) mode 4]);  
//critical section  
release (m[i]); release (m[(i+1)mod 4]);
```

Khi thực thi các quá trình trên sẽ có thể gây nên:

- A. Thrashing
- B. Deadlock
- C. Starvation
- D. Không câu nào đúng

(L.O.1.2) Thread có tất cả các tính chất sau ngoại trừ

- A. Trên máy tính đa xử lý, các thread của một process có thể được chạy trên các processor khác nhau.
- B. Trong mô hình many-to-one, khi một thread bị block thì các thread khác của process vẫn có thể tiếp tục chạy.
- C. Các thread trong cùng một process có thể truy cập dữ liệu toàn cục của process.
- D. Chi phí khởi tạo một user thread nhỏ hơn chi phí khởi tạo một kernel thread.

(L.O.1.2) Để tránh điều kiện truy đuổi (race condition), số quá trình đồng thời được thực thi trong vùng tranh chấp tại cùng một thời điểm là:

- A. 8 **B. 1** C. 16 D. 0

(L.O.1.2) Bất lợi của spinlock là gì?

- A. Không thể dùng cho nhiều hơn 2 quá trình.

B. Sử dụng busy waiting nên không thể chuyển quyền thực thi cho quá trình khác ngay cả khi không thoả mãn điều kiện vào vùng tranh chấp.

- C. Không có độ tin cậy trong một vài trường hợp.

- D. Quá phức tạp để lập trình.

(L.O.1.2) Bước nào sau đây không tồn tại trong quá trình tạo một thread mới bởi một process đang chạy?

- A. Gán định danh (thread ID) cho thread

B. Khởi tạo bảng các file đang mở (list of open files)

- C. Khởi tạo vùng nhớ stack

- D. Khởi tạo program counter

(L.O.1.2) Ưu điểm của việc giao tiếp giữa các process theo cơ chế shared-memory so với cơ chế message-passing là

- A. Thích hợp cho việc giao tiếp giữa các process trên các máy tính khác nhau

- B. Không phải giải quyết vấn đề đồng bộ, tranh chấp

C. Tốc độ cao hơn khi các process ở trên cùng một máy tính

- D. Dễ hiển thực hơn

(L.O.1.2) Nếu việc tạo quá trình thành công, thì kết quả in trên màn hình là:

```
int x = 10;
int main()
{
    int child_id;
    if ((child_id = fork()) == 0)
    {
        x++;
        printf("x = %d ", x);
    }
    else if (child_id > 0)
    {
        wait(NULL);
        x--;
        printf("x = %d ", x);
    }
    else
    {
        printf("fork error !\n");
        exit(1);
    }
    return 0;
}
```

A. x = 11 x = 9

B. x = 9 x = 11

C. x = 9 x = 10

D. x = 11 x = 10

Cho đoạn chương trình sau và giả sử việc tạo quá trình con thành công:

```
int main(){
    if (fork()==0){
```

```

        fork();
    }
    else {
        fork();
    }
    fork();
    return 0;
}

```

Cho biết số quá trình được tạo ra khi thực hiện đoạn chương trình trên.

- A. 5 **B. 8** C. 12 D. 16

(L.O.2.2) Trong các cơ chế sau, cơ chế cấp phát nào giúp truy cập file ngẫu nhiên nhanh nhất:

A. Cấp phát liên tục.

B. Cấp phát theo danh sách liên kết.

C. Cấp phát dùng chỉ mục.

D. Cấp phát dùng chỉ mục nhiều mức.

Ba (03) câu kế tiếp dùng chung mô tả dưới đây.

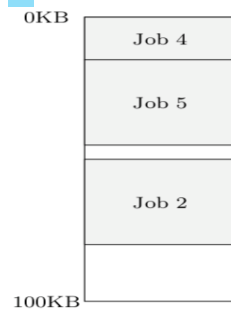
Cho một danh sách các job có thời gian bắt đầu, kết thúc và yêu cầu về bộ nhớ như bảng bên dưới.

Job	Bộ nhớ yêu cầu (KB)	Thời gian bắt đầu (phút)	Thời gian kết thúc (phút)
Job 1	50	0	20
Job 2	30	10	70
Job 3	25	15	30
Job 4	15	45	60
Job 5	30	50	70

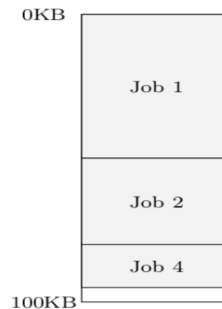
Giả thiết bộ nhớ máy tính chạy những job trên chỉ có kích thước 100KB và được quản lý theo dạng cấp phát liên tục (contiguous memory allocation) với phân hoạch kích thước thay đổi (variable-partition). Bộ nhớ chưa sử dụng sẽ được cấp phát từ thấp đến cao theo cơ chế phù hợp đầu tiên (first-fit strategy).

(L.O.1.3) Tại phút 55, hình nào sau đây thể hiện bộ nhớ máy tính? Chú ý: bộ nhớ đang được sử dụng sẽ có màu xám.

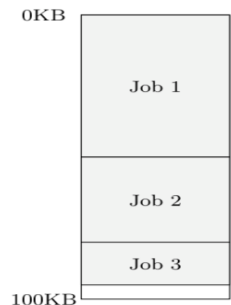
A.



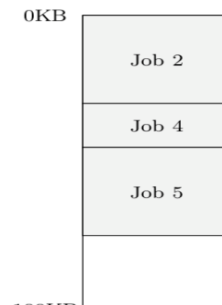
B.



C.



D.



(L.O.1.3) Một job thứ 6 đến ở phút 55, kết thúc tại phút 70 và yêu cầu 25KB bộ nhớ. Hệ điều hành có vùng bộ nhớ rảnh để cấp phát cho job này không?

- A. Có.
- B. Không.
- C. Không. Tổng bộ nhớ rảnh không đủ.
- D. Không. Có những khoảng trống (hole) rảnh nhưng kích thước của chúng < 25KB.

(L.O.1.3) Tại phút thứ 15 khi job 3 đến, bộ nhớ rảnh là không đủ cho job đó. Người ta gọi hiện tượng đó là gì?

- A. Không đủ bộ nhớ.
- B. Phân mảnh nội (internal fragmentation).
- C. Phân mảnh ngoại (external fragmentation).
- D. Tất cả các câu khác đều sai.

(L.O.1.3) Phát biểu nào sau đây là không đúng với cơ chế phân đoạn bộ nhớ ?

- A. Bộ nhớ dành cho một quá trình là một bộ những khối bộ nhớ có kích thước thay đổi.
- B. Phân mảnh ngoài không thể xảy ra.
- C. Phân mảnh ngoài có thể xảy ra.
- D. Không có quy định về thứ tự của những phân đoạn trong bộ nhớ chính.

(L.O.1.3) Xét một hệ quản lý bộ nhớ bởi cơ chế phân trang, sử dụng chuỗi địa chỉ luận lý (logical address) gồm $m = 5$ bit, trong đó phần dành cho trang là $n = 2$ bit. Cho trước bảng phân trang của một quá trình là như sau:

Page	Frame
0	5
1	6
2	10
3	2

Vậy khi quá trình truy xuất đến địa chỉ luận lý 25 (hệ thập phân) thì tương ứng với địa chỉ vật lý (hệ thập phân) nào?

A. 25.	B. 81.	C. 20.	D. 17.
--------	--------	--------	--------

(L.O.1.3) Cơ chế nào sau đây được sử dụng với hệ điều hành 64 bit ?

- A. Phân trang dạng phân cấp (hierarchical paging).
- B. Phân trang dạng nghịch đảo (inverted paging).
- C. Phân trang dạng băm (hashed paging).
- D. Phân đoạn.

(L.O.1.3) Quá trình gán địa chỉ (address binding) từ địa chỉ dạng ký hiệu (symbolic) sang địa chỉ dạng tái định vị (relocatable) được thực hiện lúc nào ?

- A. Lúc biên dịch chương trình.
- B. Lúc thực thi chương trình.
- C. Lúc tải chương trình vào bộ nhớ để thực thi.
- D. Lúc đoạn chương trình liên quan được thực thi bởi bộ xử lý.

(L.O.1.3) Trong cơ chế phân trang có sử dụng bộ nhớ ảo, khi không còn frame chưa sử dụng để cấp phát cho một yêu cầu bộ nhớ của một quá trình thì hệ điều hành sẽ thực hiện điều nào sau đây?

- A. Kết thúc (terminate) quá trình yêu cầu bộ nhớ.
- B. Đưa quá trình khỏi bộ nhớ chính và chuyển quá trình sang trạng thái waiting.
- C. Kích hoạt cơ chế thay thế trang (page replacement).
- D. Tất cả các tác vụ nêu trong những câu khác.

(L.O.1.3) Chuỗi con tham khảo (reference string) bộ nhớ 0, 1, 2, 3, 4 được thực hiện lại 50 lần kế tiếp nhau. Hỏi nếu sử dụng cơ chế thay thế trang FIFO với bộ nhớ chính 4 frame thì sẽ có bao nhiêu lần thay thế trang ? (Giả thiết ban đầu cả 4 frame đều chưa sử dụng).

- A. 50. **B. 246.** C. 200. D. 62.

(L.O.1.1) Thuật ngữ nào đề cập đến khả năng của một hệ điều hành hỗ trợ nhiều luồng thực thi với một tiến trình duy nhất?

- A. Đa luồng (Multithreading).**
B. Đa xử lý (Multiprocessing).
C. Đa nhiệm (Multiexecuting).
D. Song luồng (Bi-threading).

(L.O.1.2) là khả năng của nhiều tiến trình phối hợp các hoạt động của chúng bằng cách trao đổi thông tin.

- A. Đồng bộ hóa (Synchronization)**
B. Loại trừ lẫn nhau (Mutual Exclusion)
C. Trạng thái đóng chết (Deadlock)
D. Đói (Starvation)

(L.O.1.2) Kỹ thuật có thể được sử dụng để giải quyết xung đột, chẳng hạn như cạnh tranh tài nguyên và đồng bộ hóa các quá trình để chúng có thể hợp tác.

- A. Loại trừ lẫn nhau (Mutual Exclusion).
B. Chờ đợi bận rộn (Busy Waiting).
C. Bế tắc (Deadlock).
D. Đói (Starvation).

(L.O.1.2) Đồng bộ (synchronization) khi làm việc với các thread là cần thiết vì:

- A. Các thread chia sẻ cùng không gian địa chỉ
B. Các thread chia sẻ biến toàn cục
C. Các thread có thể chỉ chia sẻ 1 file
D. Tất cả các ý trên.

(L.O.1.3) Cho dữ liệu sau. Địa chỉ luận lý (3, 40) có địa chỉ vật lý tương ứng là:

Segment	Base	Length
0	219	600
1	2300	14
2	90	100
3	1400	500
4	1950	100

- A. 130. **B. 1440.**
C. 1990. D. Không hợp lệ.

(L.O.2.1) Giả sử hệ thống có 4 quá trình, với thời điểm đến và CPU burst như trong bảng dưới.

Process	Thời gian đến	CPU burst
A	0	8
B	2	5
C	3	2
D	4	3

(L.O.2.1) Nếu dùng giải thuật định thời Shortest Job First, thời gian đợi trung bình của các quá trình sẽ là:

- A. 4 **B. 5,5** C. 7,5

D. 6,5

(L.O.2.1) Nếu dùng giải thuật định thời Shortest Remaining Time First, thời gian đợi trung bình của các quá trình sẽ là:

- A. 4,0 B. 5,5 C. 7,5
D. 5,0.

(L.O.2.1) Nếu dùng giải thuật định thời Round Robin với quantum=2, thời gian đợi trung bình của các quá trình sẽ là:

- A. 4,0 B. 5,5 C. 7,5
D. Không câu nào đúng

(L.O.1.3) Cho các yêu cầu về bộ nhớ của các job 1-4 và các khối bộ nhớ còn trống. Nếu First-fit được chọn thì job nào không được thực thi?

Jobs: J1 15K, J2 20K, J3 20K, J4 50K

Free Blocks: B1 30K, B2 15K, B3 50K, B4 20K

- A. J1 B. J2 C. J3 D. J4

(L.O.1.1) Chọn phát biểu đúng về cấu trúc monolithic:

- A. Có hiệu năng (performance) giảm so với các cấu trúc khác.
B. Việc trao đổi dữ liệu giữa các thành phần chức năng của kernel thực hiện chủ yếu bằng cơ chế message-passing.
C. Các dịch vụ của hệ điều hành đều được tích hợp vào kernel.
D. Việc thêm các chức năng của hệ điều hành được thực hiện dễ dàng hơn so với cấu trúc microkernel.

(L.O.1.1) Giao diện (Interface) cung cấp để truy cập các dịch vụ của hệ điều hành thông qua:

- A. Thư viện hàm (library).
B. Gọi hệ thống (System calls).
C. Các lệnh mã máy (assembly instructions).
D. Application Programming Interface (API).

(L.O.2.2) Một partition đĩa có các block còn trống gồm có 2, 3, 4, 5, 8, 9, 10, 11, 12, 13, 17, 18, 25, 26 và 27. Nếu sử dụng bitmap để quản lý không gian trống thì chuỗi bitmap của partition trên là? (Giả sử nếu block trống bit là 1, ngược lại bit là 0)

- A. 0000110000001110011111100011111...
B. 110000110000001110011111100011111..
C. 1111001111110001100000011100000...
D. 001111001111110001100000011100000...

(L.O.2.1) Giải thuật định thời CPU nào dưới đây phù hợp cho các ứng dụng tương tác?

- A. Shortest remaining time first.
B. RR (Round Robin).
C. SJF (Shortest Job First)
D. FCFS (First Come First Served).

(L.O.1.1) Hệ thống đa lập trình (Multiprogramming system):

- A. là hệ thống dễ phát triển hơn hệ thống đơn lập trình.
B. thực hiện mỗi công việc nhanh hơn.
C. chỉ được dùng trong các hệ thống máy tính lớn trước đây.
D. thực hiện nhiều công việc cùng lúc đưa đến hiệu suất hệ thống tốt hơn.

(L.O.1.2) So sánh các kỹ thuật đồng bộ:

- A. Đối với kỹ thuật dùng busy-waiting, trong khi chờ được thực thi vùng tranh chấp, process sẽ tạm thời chuyển sang trạng thái block.
- B. Đối với kỹ thuật không busy-waiting, khi một process được wake up, nó sẽ lập tức chuyển từ trạng thái waiting sang running.
- C. So với kỹ thuật không busy-waiting, kỹ thuật dùng busy-waiting giúp giảm việc chuyển ngữ cảnh của các process liên quan.
- D. Kỹ thuật không busy-waiting sử dụng CPU không hiệu quả bằng kỹ thuật dùng busy-waiting.

(L.O.1.3) Trong quản lý bộ nhớ, ưu điểm của phân chia động so với phân chia cố định là:

- A. Giảm phân mảnh ngoại.
- B. Giảm phân mảnh nội.
- C. Giảm phân mảnh nội và phân mảnh ngoại.
- D. Không gây ra phân mảnh.

(L.O.1.3) Cho thứ tự của chuỗi tham chiếu trang như sau: 2 3 6 4 6 3 1 2 4 6. Với số lượng frame được cấp là 3 frames trống, hãy tính số lần page fault nếu (câu này bị sai đáp án đề gốc):

Dùng giải thuật thay trang FIFO:

- A. 5 B. 6 C. 7 D. 8

Dùng giải thuật thay trang OPTIMAL:

- A. 5 B. 6 C. 7 D. 8

Dùng giải thuật thay trang LRU:

- A. 5 B. 6 C. 7 D. 8

(L.O.1.2) Có bao nhiêu semaphore được dùng trong bài toán Producer – Consumer?

- A. 1 B. 2 C. 3 D. 4

(L.O.1.2) Chọn phát biểu đúng về process và thread:

- A. Các thread có thể đọc ghi trong cùng không gian bộ nhớ còn các process thì không.
- B. Các thread có chung vùng stack còn các process thì có vùng stack riêng biệt.
- C. Các process có thể giao tiếp với nhau còn các thread thì không.
- D. Cả A, B và C đều sai.

(L.O.1.2) Cơ chế loại trừ tranh chấp nào sau đây không có busy-waiting:

- A. Giải thuật của Peterson.
- B. TestAndSet (hay TSL).
- C. Semaphore.
- D. Compare-and-swap.

(L.O.1.2) Cơ chế đồng bộ nào sau đây có hiệu dụng CPU tốt nhất:

- A. Sử dụng giải thuật Peterson.
- B. Lệnh đơn nguyên TestandSet.
- C. Sử dụng giải thuật Dekker.
- D. Sử dụng Semaphore.

(L.O.2.1) Chiến lược định thời nào cho phép những process đang thực thi có thể tạm ngưng (temporarily suspended)?

A. Shortest remaining time first.

B. Non-preemptive scheduling.

C. Shortest job first.

D. First Come First Served.

(L.O.2.2) Loại cấp phát khối dữ liệu nào trên đĩa mà File Control Block của nó chứa khối đầu và số khối dành cho tập tin ?

A. Cấp phát dạng liên kết (linked allocation)

B. Cấp phát dạng chỉ mục (indexed allocation)

C. Cấp phát dạng liên tục (contiguous allocation)

D. Cấp phát dạng băm (hash allocation)

(L.O.2.2) Xét một hệ thống tập tin có những khối dữ liệu trên đĩa được đánh số từ 0-49. Hệ thống tập tin sử dụng phương pháp cấp phát liên tục (contiguous allocation), với bảng thư mục như dưới đây:

Tập tin	Bắt đầu	Chiều dài
abc.txt	8	15
xyz.dll	27	14
mnp.exe	43	7

Hệ điều hành được yêu cầu tạo một tập tin mới với kích thước gồm 9 khối. Hỏi có thể tạo được tập tin đó trên hệ thống tập tin đang dùng hay không và nếu có thì ở có thể ở đâu?

A. Có thể và bắt đầu từ khối 0.

B. Có thể và bắt đầu từ khối 22.

C. Có thể và bắt đầu từ khối 41.

D. Không thể được.

-----HẾT-----