



## ON TAP HE IEU HANH - hihhi

Hệ điều hành (Đại học Tôn Đức Thắng)



Scan to open on Studocu

## ÔN TẬP HỆ ĐIỀU HÀNH

### \*Chương 6+7

#### Question 1

0 / 1 pts

Đoạn mã nào trong các tiến trình có thể gây ra lỗi khi được thực thi đồng thời?

Exit Section.

Critical Section.(correct )

Remainer Section.

Entry Section.

#### Question 2

0 / 1 pts

Đồng bộ hoá (Process Synchronization) là công việc cần phải áp dụng cho loại tiến trình nào?

Tiến trình cộng tác (Cooperating process).(correct )

Tiến trình độc lập (Independent process)

Tiến trình người dùng (User process).

Tiến trình hệ thống (System process).

#### Question 3

0 / 1 pts

Đoạn mã nào được sử dụng để kiểm soát quá trình đồng bộ?

Critical section.

Entry section.(correct answer )

Remainder section.

Program code.

#### Question 4

0 / 1 pts

Đoạn mã nào có thể chạy cùng lúc mà không gây ra sai sót dữ liệu?

Program code

Entry section.

Critical section.

Remainder section.(correct answer)

#### Question 5

0 / 1 pts

Biến số đơn nguyên (atomic variable) là gì?

Các thao tác lên biến số này tuân tự được thực thi trong CPU. (**Correct Answer**)

Biến số chỉ chứa duy nhất một kiểu dữ liệu được định nghĩa trước.

Biến số chỉ có ý nghĩa địa phương, sử dụng nội bộ trong tiểu trình.

Các thao tác lên biến số này được song song thực hiện trong CPU.

#### Question 6

0 / 1 pts

Giải thuật Peterson sử dụng các biến số điều khiển nào để giải quyết bài toán đồng bộ giữa hai tiến trình?

boolean flag[2];

int sync = 2

boolean flag[2] và int turn;(Correct Answer)

choosing[i] = true và int number[i];

### Question 7

0 / 2 pts

Một tiến trình Px thực hiện thao tác `signal()` trên một biến số Semaphore n thì có tác dụng gì?

n++ và sau đó nếu  $n > 0$  thì `wake_up()` một tiến trình đang bị blocked.

n++ và sau đó nếu  $n > 0$  thì `wake_up()` tiến trình Px.

n++ và sau đó nếu  $n \leq 0$  thì `wake_up()` tiến trình đang bị blocked. **(Correct Answer)**

n++ và sau đó nếu  $n \leq 0$  thì `wake_up()` tiến trình Px.

### Question 8

0 / 2 pts

Một tiến trình Px thực hiện thao tác `wait()` trên một biến số Semaphore n thì có tác dụng gì?

n-- và sau đó nếu  $n \geq 0$  thì `block()` tiến trình Px.

n-- và sau đó nếu  $n < 0$  thì `block()` tiến trình Px. **(Correct Answer)**

n-- và sau đó nếu  $n \leq 0$  thì `block()` các tiến trình khác Px.

n++ và sau đó nếu  $n \leq 0$  thì `block()` tiến trình Px.

### Question 9

0 / 2 pts

Giải thuật / Phương pháp nào sau đây chỉ có thể giải quyết đồng bộ không nhiều hơn 2 tiến trình?

Phương pháp Semaphore.

Giải thuật Banker.

Giải thuật Peterson. **(Correct Answer)**

Phương pháp Hàng rào bộ nhớ.

### Question 10

0 / 2 pts

Tình trạng cạnh tranh (Race condition) là gì?

Tiến trình không cho phép các tiến trình khác tác động lên biến số của nó, và dẫn đến việc đồng bộ thất bại.

Người sử dụng yêu cầu chạy 02 tiến trình có tranh chấp dữ liệu, gây nên hiện tượng tắc nghẽn cho hệ thống.

Khi nhiều hơn một tiến trình thao tác lên dữ liệu chia sẻ, kết quả cuối cùng phụ thuộc vào thứ tự thực thi của các thao tác đó. (**Correct Answer**)

Các lệnh cấp thấp (là mã máy) được thực thi đồng thời trong một chu kỳ lệnh của CPU làm sai sót dữ liệu.

### Question 11

0 / 2 pts

Kỹ thuật đồng bộ sử dụng Semaphore giải quyết được vấn đề gì mà giải thuật Peterson chưa làm được?

Progress (Tính tiến triển).

Bounded-Waiting (Chờ vô hạn định).

Busy-waiting (Chờ đợi bận rộn). (**Correct Answer**)

Mutual Exclusion (Loại trừ tương hỗ).

### Question 12

0 / 2 pts

Yêu cầu về tính sống còn (liveness) của các giải pháp đồng bộ đảm bảo điều gì cho hệ thống?

Hệ thống đang xử lý các tiến trình có hiệu năng khai thác cao

Các tiến trình luôn tiến triển, tài nguyên không cạn kiệt. (**Correct Answer**)

Sự chờ đợi bận rộn (Busy waiting) không xuất hiện với mọi tiến trình.

Dữ liệu luôn được đồng bộ và không có sai sót khi cập nhật.

### Question 13

0 / 2 pts

**Mục đích của việc sử dụng Semaphore là gì?**

Trị số của Semaphore cho biết process nào đang được thực thi.

Thông tin của Semaphore phục vụ cho bài toán đồng bộ tiến trình(**Correct Answer**).

Trị số của Semaphore cho biết số tiến trình tối đa được vào hệ thống.

Semaphore là tín hiệu ngắt gửi cho hệ điều hành khi cần đồng bộ tiến trình.

### Question 14

0 / 2 pts

Phương pháp Hàng rào bộ nhớ (Memory Barrier) được hiện thực ra sao?

Các tiến trình được cấp các bản sao vùng nhớ chia sẻ để thao tác cập nhật.

Việc cập nhật vùng nhớ chia sẻ được quyết định bởi tiến trình cấp phát hàng rào.

Các câu lệnh thay đổi biến số chia sẻ cần được nhìn thấy bởi mọi tiến trình khác.(**Correct Answer**)

Các vùng nhớ chia sẻ cần được nhìn thấy bởi tất cả tiến trình đang đồng bộ.

### Question 15

0 / 2 pts

“Critical Section” mô tả đoạn mã như thế nào trong một tiến trình?

Đoạn mã có yêu cầu tính toán và sử dụng toàn bộ CPU.

Đoạn mã có yêu cầu nhập xuất dữ liệu từ thiết bị ngoại vi.

Đoạn mã có chứa những thao tác lên biến dùng chung. **(Correct Answer)**

Đoạn mã hệ điều hành tự thêm vào trong tiến trình.

### Question 16

0 / 2 pts

“Entry / Exit Section” là đoạn mã gì?

Đoạn mã có chứa những thao tác lên biến dùng chung.

Đoạn mã hệ điều hành thêm vào trước và sau đoạn mã nguy cơ (Critical section). **(Correct Answer)**

Đoạn mã có yêu cầu tính toán và sử dụng toàn bộ CPU.

Đoạn mã có chứa lệnh can thiệp vào hoạt động của hệ điều hành.

### Question 17

0 / 3 pts

Cho hai tiến trình P1 và P2 quyền tác động lên biến semaphore chia sẻ S và Q (đều có khởi tạo = 1). Các lệnh sau đây lần lượt được thực thi, hệ thống sẽ diễn tiến như thế nào?

| Time  | P1        | P2        |
|-------|-----------|-----------|
| t = 1 | wait(S)   |           |
| t = 2 |           | wait(Q)   |
| t = 3 | wait(Q)   |           |
| t = 4 |           | wait(S)   |
| t = 5 | signal(S) |           |
| t = 6 |           | signal(Q) |
| t = 7 | signal(Q) |           |
| t = 8 |           | signal(S) |

Hệ thống sẽ đảm bảo P2 hoàn tất trước P1.

Hệ thống sẽ chạy hết tất cả lệnh đã nêu.

Hệ thống sẽ đảm bảo P1 hoàn tất trước P2.

Hệ thống sẽ rơi vào trạng thái Deadlock. **(Correct Answer)**



### Question 18

0 / 3 pts

Cho đoạn mã của 2 tiến trình như sau:

**P1:**

*(các lệnh khác)*

**wait(mutex);**

**critical section**

**signal(mutex);**

*(các lệnh khác)*

**P2:**

*(các lệnh khác)*

**wait(mutex);**

**critical section**

**signal(mutex);**

*(các lệnh khác)*

Trong đó biến mutex là biến toàn cục dùng chung (shared variable). Phát biểu nào sau đây là đúng với hệ thống nêu trên?

---

Với khởi tạo mutex = 1; chỉ có 1 tiến trình được vào critical section. (**Correct Answer** )

---

Với khởi tạo mutex = 0; chỉ có 1 tiến trình được vào critical section.

---

Với khởi tạo mutex = 1; P2 phải gửi tín hiệu đến P1 để xin vào critical section.

---

Với khởi tạo mutex = 2; P2 chắc chắn sẽ vào critical section trước.

### Question 19

0 / 3 pts

Cho đoạn mã của 2 tiến trình P1 và P2 như sau:

**P1:**

*(các lệnh khác)*

**signal(mutex);**

**func\_1( );**

*(các lệnh khác)*

**P2:**

*(các lệnh khác)*

**wait(mutex);**

**func\_2( );**

*(các lệnh khác)*

Trong đó biến mutex là biến toàn cục dùng chung (Shared variable)  
Chọn phát biểu đúng.

---

Để đảm bảo hàm func\_2() chạy trước func\_1(), khởi tạo mutex = 0.

---

Để đảm bảo hàm func\_2() chạy trước func\_1(), khởi tạo mutex = 2.

---

Để đảm bảo hàm func\_1() chạy trước func\_2(), khởi tạo mutex = 1.

---

### Correct Answer

Để đảm bảo hàm func\_1() chạy trước func\_2(), khởi tạo mutex = 0.

### Question 20

0 / 3 pts

Semaphore được hiện thực như thế nào?

Mảng các số nguyên hoặc nhị phân, kèm theo 2 thao tác block() và wake\_up().

### Correct Answer

Biến số nguyên hoặc nhị phân, kèm theo 2 thao tác wait() và signal().

Biến số nguyên hoặc nhị phân, kèm theo 2 thao tác block() và wake\_up().

Mảng các số nguyên hoặc nhị phân, kèm theo 2 thao tác wait() và signal().

### Question 21

0 / 1 pts

Bài toán “Bộ đệm giới hạn” (Bounded Buffer) đề cập đến vấn đề chính yếu gì?

Khoá chặn truy cập chỉ của một vài tiến trình đang thực thi.

Bảo mật thông tin khi gửi và nhận thông điệp giữa các tiến trình.

### Correct Answer

Gửi và nhận gói tin qua bộ nhớ chia sẻ có kích thước nhất định.

Tranh chấp tài nguyên giữa nhiều tiến trình trong lúc thực thi.

### Question 22

0 / 1 pts

Bài toán “Bộ ghi – Bộ đọc” (Writers and Readers) đề cập đến vấn đề chính yếu gì?

#### Correct Answer

Dữ liệu chia sẻ mà chỉ một vài tiến trình mới có nhu cầu cập nhật dữ liệu.

Phân phối dữ liệu từ nhiều tiến trình nguồn đến nhiều tiến trình đích.

Bảo mật thông tin khi chia sẻ thông tin giữa các tiến trình đang thực thi.

Gửi và nhận gói tin qua bộ nhớ chia sẻ có kích thước nhất định.

### Question 23

0 / 1 pts

Bài toán “Triết gia ăn tối” (Dining Philosophers) đề cập đến vấn đề chính yếu gì?

Tranh chấp các tài nguyên chia sẻ riêng biệt giữa từng cặp tiến trình(**Correct Answer**).

Chia sẻ tài nguyên thành nhiều thực thể để đáp ứng cho nhiều tiến trình.

Bảo mật thông tin chia sẻ thông tin giữa nhiều tiến trình với nhau.

Hiệu suất sử dụng tài nguyên trong hệ thống chạy song song nhiều tiến trình.

### Question 24

0 / 2 pts

Bài toán “Bộ đệm giới hạn” (Bounded Buffer) có thể giải quyết bằng bao nhiêu biến số semaphore?

Chỉ cần 2 biến: full và empty.

1 mảng sem[5].

Duy nhất biến số n.

### Correct Answer

3 biến: mutex, full và empty.

### Question 25

0 / 2 pts

Bài toán “Bộ ghi – Bộ đọc” (Writers and Readers) có đặc trưng gì?

Hệ thống chỉ có một bộ ghi và rất nhiều bộ đọc.

### Correct Answer

Các bộ đọc mới có thể cập nhật dữ liệu chia sẻ.

Tất cả bộ đọc và bộ ghi cần xếp hàng để thực thi.

Hệ thống chỉ có một bộ đọc và rất nhiều bộ ghi.

### Question 26

0 / 2 pts

Bài toán “Triết gia ăn tối” (Dining Philosophers) nếu sử dụng semaphore thì chúng được khởi tạo như thế nào?

semaphore chopstick[5], tất cả phần tử gán bằng 1. (**Correct Answer** )

semaphore chopstick, khởi tạo giá trị 5.

semaphore chopstick[5], tất cả phần tử gán bằng 1.

semaphore chopstick[5], các phần tử gán lần lượt từ 1 đến 5.

### Question 27

0 / 2 pts

API POSIX cung cấp nhiều công cụ đồng bộ, nhưng không bao gồm công cụ nào sau đây?

Biến số semaphore

Khoá mutex lock

Dispatcher objects. (**Correct Answer** )

Biến số điều kiện (condition variable).

### Question 28

0 / 1 pts

Bài toán “Bộ ghi – Bộ đọc” (Writers and Readers) các biến số được khởi tạo như thế nào?

semaphore rw\_mutex = 0, mutex = 1; int read\_count = 0;

semaphore rw\_mutex = 1, mutex = 1; int read\_count = 0; **(Correct Answer)**

semaphore rw\_mutex = 1, mutex = 2; int read\_count = 0;

semaphore rw\_mutex = 1, mutex = 1; int read\_count = 2;

### Question 29

0 / 3 pts

Bài toán “Triết gia ăn tối” (Dining Philosophers) có thể giải quyết bằng phương pháp nào để tránh bị tắc nghẽn (deadlock)?

Giải thuật Peterson với các vòng lặp kiểm tra while().

Các biến số semaphore với các lệnh wait() và signal().

Bộ quan sát (Monitor) với các lệnh test(). **(Correct Answer)**

Các khoá mutex\_lock áp dụng cho từng vùng tranh chấp.

### Question 30

0 / 3 pts

Bài toán “Bộ ghi – Bộ đọc” (Writers and Readers) có biến thể thứ 2, nó khác gì với biến thể đầu tiên?

---

Nếu một bộ ghi mới đến, nó sẽ được thực thi sớm nhất có thể. **(Correct Answer)**

Các bộ đọc có thể thực thi song song mà không sai sót dữ liệu.

Số lượng bộ đọc và bộ ghi bị giới hạn để tránh cạn kiệt tài nguyên.

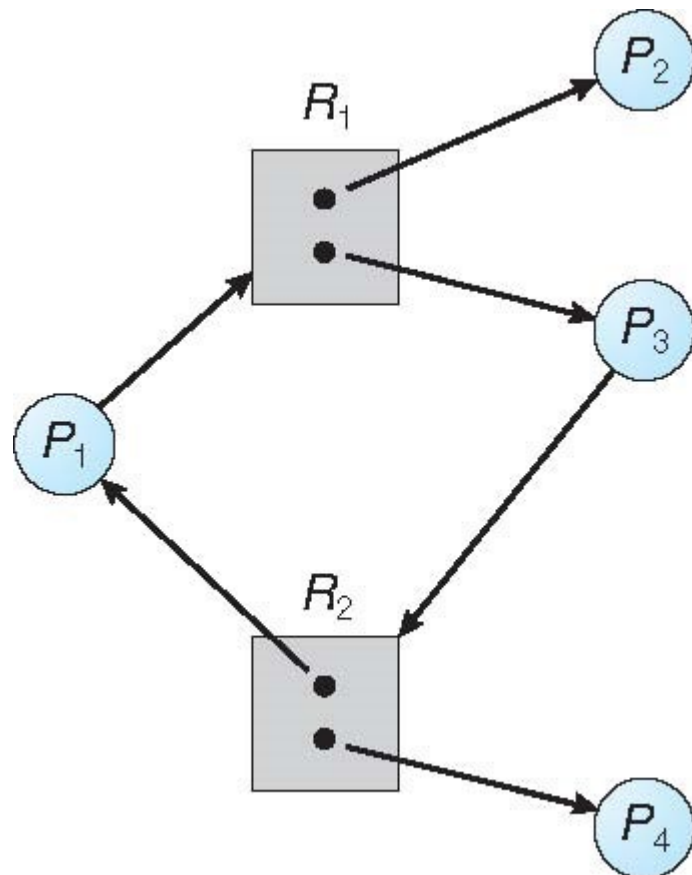
Nếu một bộ đọc mới đến, nó sẽ được thực thi sớm nhất có thể.



## \*chương 8

### Question 1

0 / 1 pts



Đồ thị phía trên được gọi là đồ thị gì?

---

#### Correct Answer

Resource Allocation Graph.

Request and Allocation

Resource and Process.

Request Allowed Graph.

## Question 2

0 / 1 pts

Deadlock là trạng thái như thế nào?

Mọi tiến trình đều đang yêu cầu sử dụng CPU.

Hệ thống không thể tiếp tục hoạt động

Các tiến trình không thể tiếp tục được thực thi. **Correct Answer**

Hệ điều hành đóng băng các tiến trình.

## Question 3

0 / 1 pts

Đồ thị RAG của một hệ thống mô tả cho  $n$  tiến trình và  $m$  tài nguyên thì có bao nhiêu đỉnh (lực lượng của tập đỉnh  $V$ )?

$n * m$

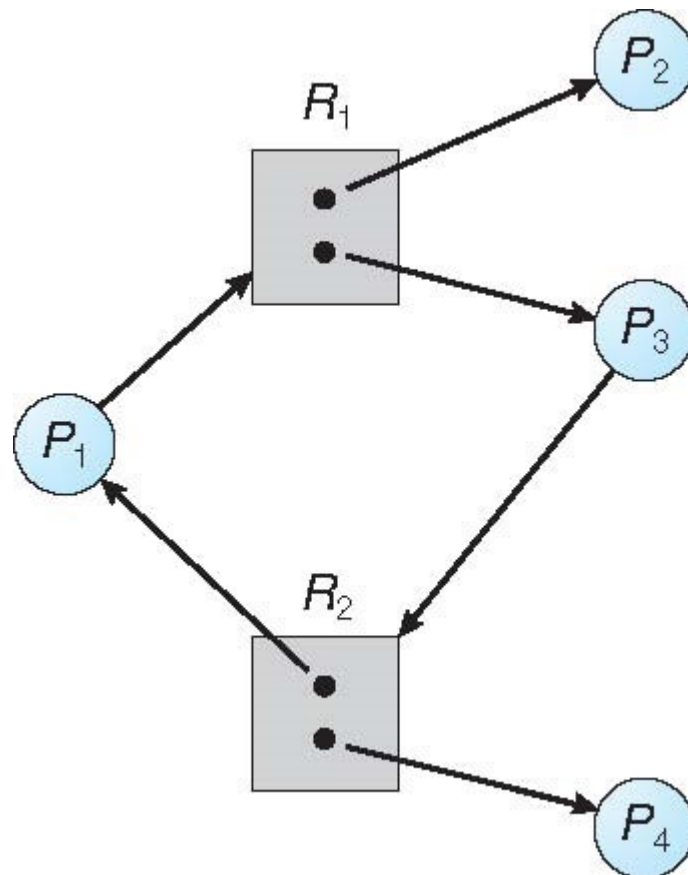
$n + m$  **Correct Answer**

tối thiểu là  $n + m$

$n$

#### Question 4

0 / 1 pts



Đồ thị phía trên thể hiện thông tin gì?

Tiến trình P1 đã có đủ tài nguyên nó cần.

---

#### Correct Answer

Hệ thống có 4 tiến trình và 2 loại tài nguyên.

Hệ thống có 4 tiến trình và 2 loại thực thể.

Tiến trình P1 đã sẵn sàng vào CPU.

### Unanswered Question 5

0 / 1 pts

Mục tiêu của giải thuật “Nhà băng” (Banker) là gì?

Tìm ra những chuỗi không an toàn trong hệ thống để phòng trừ.

Tìm ra thứ tự nạp vào hệ thống các chương trình mà người dùng yêu cầu.

Chỉ ra một thứ tự thực thi các tiến trình sau khi hệ thống bị deadlock.

#### Correct Answer

Chỉ ra một thứ tự thực thi của các tiến trình sao cho hệ thống luôn an toàn.

### Question 6

0 / 1 pts

Trạng thái mà một hệ thống máy tính có các tiến trình vẫn hoạt động nhưng thời gian đáp ứng rất lâu là gì?

Low resource

Starvation **Correct Answer**

Deadlock

Unsafe

### Unanswered Question 7

0 / 2 pts

Quan hệ giữa “an toàn” và “deadlock” được diễn đạt như thế nào?

Hệ thống vẫn có thể bị deadlock khi nó đang an toàn.

An toàn và deadlock là 2 khái niệm cùng chỉ 1 trạng thái

#### Correct Answer

Hệ thống chỉ có thể bị deadlock khi nó có trạng thái không an toàn.

Hệ thống sẽ bị deadlock khi nó có trạng thái không an toàn.

#### Unanswered Question 8

0 / 2 pts

Cho hệ thống có 5 tiến trình và 4 loại tài nguyên: A, B, C và D. Áp dụng giải thuật Banker (Nhà băng).

| Process        | Allocation |   |   |   | Max |   |   |   | Available |   |   |   |
|----------------|------------|---|---|---|-----|---|---|---|-----------|---|---|---|
|                | A          | B | C | D | A   | B | C | D | A         | B | C | D |
| P <sub>0</sub> | 0          | 0 | 1 | 2 | 0   | 0 | 1 | 2 | 1         | 5 | 2 | 0 |
| P <sub>1</sub> | 1          | 0 | 0 | 0 | 1   | 7 | 5 | 0 |           |   |   |   |
| P <sub>2</sub> | 1          | 3 | 5 | 4 | 2   | 3 | 5 | 6 |           |   |   |   |
| P <sub>3</sub> | 0          | 6 | 3 | 2 | 0   | 6 | 5 | 2 |           |   |   |   |
| P <sub>4</sub> | 0          | 0 | 1 | 4 | 0   | 6 | 5 | 6 |           |   |   |   |

Hệ thống này có bao nhiêu thực thể tài nguyên mỗi loại?

---

2 thực thể A, 14 thực thể B, 10 thực thể C và 12 thực thể D.

2 thực thể A, 9 thực thể B, 10 thực thể C và 12 thực thể D.

---

**Correct Answer**

3 thực thể A, 14 thực thể B, 12 thực thể C và 12 thực thể D.

---

3 thực thể A, 22 thực thể B, 21 thực thể C và 16 thực thể D.

### Unanswered Question 9

0 / 2 pts

Chọn phát biểu đúng cho điều kiện để tồn tại deadlock **no preemption** :

---

Tồn tại một chu kỳ đóng các yêu cầu tài nguyên.

Tiến trình vẫn sở hữu tài nguyên khi yêu cầu một tài nguyên khác.

Tài nguyên không thể cấp phát cho tiến trình khác.

---

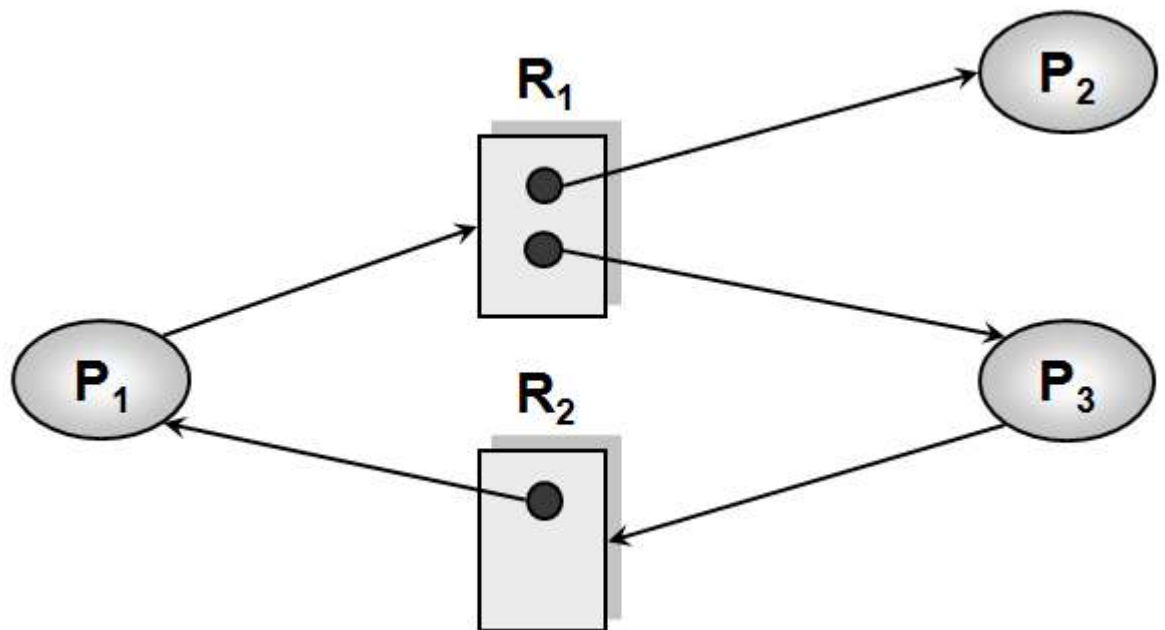
**Correct Answer**

Hệ thống không đòi lại được tài nguyên sau khi đã cấp phát.

### Unanswered Question 10

0 / 2 pts

Cho đồ thị



Theo đồ thị RAG phía trên, có thể kết luận gì cho hệ thống?

**Correct Answer**

$P_1$  chỉ có thể chạy tiếp khi  $P_2$  kết thúc

Chuỗi an toàn:  $P_2 - P_3 - P_1$

Hệ thống đã bị deadlock.

$P_3$  đang chiếm giữ một thực thể của tài nguyên  $R_2$

### Unanswered Question 11

0 / 2 pts

Thứ tự của quy trình yêu cầu cấp phát tài nguyên là:

---

**Correct Answer**

request – use – release

---

release – use – request.

---

release – request – use.

---

use – request – release.

**Unanswered Question 12**

0 / 2 pts

Yếu tố nào sau đây không phải là một đặc trưng của Deadlock?

Giữ và chờ (Hold and wait).

Loại trừ tương hỗ (Mutual Exclusion).

Không thể chiếm lại tài nguyên (No preemption).

---

**Correct Answer**

Hệ thống thiếu thốn tài nguyên (Starvation)

**Unanswered Question 13**

0 / 2 pts

Phát biểu nào sau đây SAI về đồ thị cấp phát tài nguyên



Tài nguyên có thể có nhiều thực thể

---

**Correct Answer**

Đồ thị có chu trình thì hệ thống bị deadlock

---

Đồ thị không có chu trình thì hệ thống không bị deadlock.

---

Tập đỉnh V gồm có 2 loại là tiến trình và tài nguyên

---

**Unanswered Question 14**

0 / 2 pts

Cho đồ thị hình bên dưới :

Theo đồ thị RAG phía trên, có thể kết luận gì cho hệ thống?

---

Chuỗi an toàn:  $P4 - P1 - P2 - P3$ .

---

Deadlock đang xảy ra với P1 và P3.

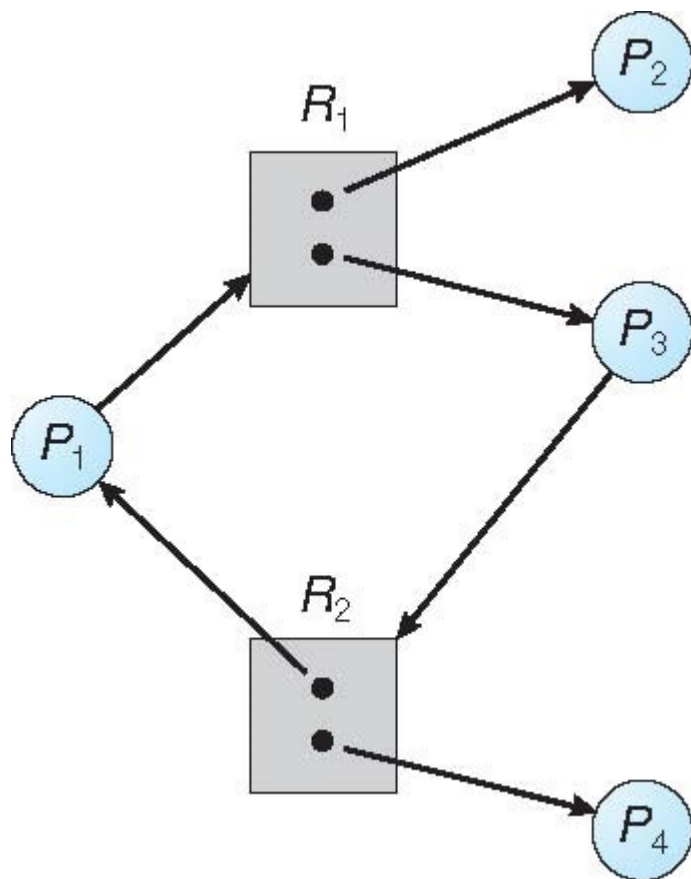
---

**Correct Answer**

P1 chỉ có thể chạy tiếp khi P2 hoặc P3 kết thúc.

---

Hệ thống đã bị tắc nghẽn.



### Unanswered Question 15

0 / 1 pts

Cho đồ thị RAG bên dưới :

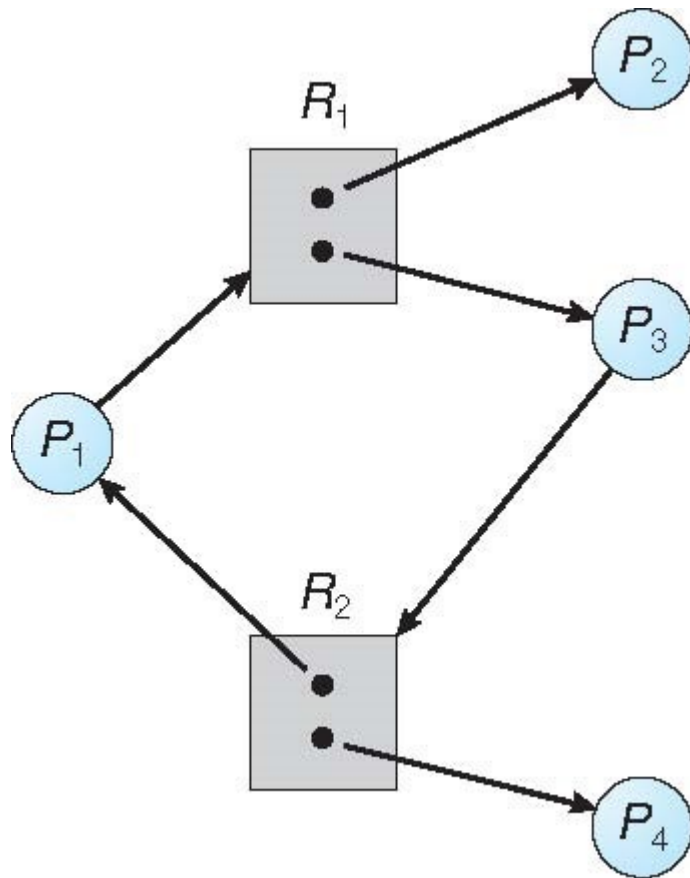
Theo đồ thị RAG phía trên, phát biểu nào ĐÚNG?

P3 đang yêu cầu thực thể đã bị P4 chiếm giữ.

P2 đang yêu cầu một thực thể của tài nguyên R1.

P1 đang chiếm giữ toàn bộ tài nguyên R1.

P1 đang chiếm giữ một thực thể của tài nguyên R2. **Correct Answer**



### Unanswered Question 16

0 / 3 pts

Cho tập cạnh E của một đồ thị RAG như sau :  $E = \{(P1,R1),(R1,P2),(P2,R2),(P3,R1),(R2,P3)\}$  chọn phát biểu đúng :

#### Correct Answer

RAG trên là RAG vòng.

RAG trên không bị deadlock..

RAG trên không bị vòng.

RAG trên bị deadlock.

### Unanswered Question 17

0 / 3 pts

Khi hệ thống xảy ra deadlock, hệ điều hành phải chọn một tiến trình (nạn nhân) để kết thúc. Tính chất nào sau đây sẽ KHÔNG được quan tâm?

Tiến trình nạn nhân là độc lập (interactive) hay theo bó (batch).

Thời gian mà tiến trình nạn nhân đã vận hành và tiếp tục cần để chạy.

Tiến trình nạn nhân cần bao nhiêu tài nguyên để có thể chạy tiếp.

Trạng thái deadlock của hệ thống là do tiến trình nào gây ra. **Correct Answer**

### Unanswered Question 18

0 / 3 pts

Một hệ thống có  $n$  tiến trình và  $m$  loại tài nguyên và đang ở trạng thái Deadlock. Lúc đó, nếu chạy giải thuật “Phát hiện deadlock” (Detection Algorithm) thì độ phức tạp là

$O(n)$ .

$O(m * n)$

**Correct Answer**

$O(m * n * n)$

$O(1)$ .

### Question 1

0 / 1 pts

Trong cơ chế phân trang bộ nhớ của hệ điều hành, khái niệm trang (Page) là gì?

Kích thước của một tiến trình cơ bản nhất trong hệ thống

Đơn vị phân hoạch trong không gian tiến trình. **Correct Answer**

Dung lượng nhớ lớn nhất mà một tiến trình có thể xin cấp phát.

Đơn vị phân hoạch trong không gian bộ nhớ vật lý.

### Question 2

0 / 1 pts

Hiện tượng “Phân mảnh nội” xảy ra với những vùng trống bộ nhớ nào?

Thừa ra do hệ điều hành cấp phát nhiều hơn yêu cầu.

Tất cả vùng trống chưa sử dụng trong bộ nhớ chính.

**Correct Answer**

Thừa ra do tiến trình xin cấp phát nhiều hơn nhu cầu thật sự

Sinh ra do sự thay đổi số lượng biến số của tiến trình theo thời gian.

### Question 3

0 / 3 pts

Thời điểm nào có thể “ánh xạ” địa chỉ chương trình vào địa chỉ bộ nhớ vật lý? Chọn mọi phương án đúng.

#### Correct Answer

Load

Swapping

#### Correct Answer

Excution

Running

Waiting

#### Correct Answer

Compile

Implement

### Question 4

0 / 1 pts

**Cấp phát bộ nhớ theo phương pháp “Phân trang” có đặc trưng nào sau đây?**

Một trang có kích thước đủ để chứa toàn bộ kích thước của tiến trình.

#### Correct Answer

Kích thước một trang (page) và một khung trang (frame) bằng nhau.

Tiến trình được cấp phát đủ số khung trang mà tiến trình đó yêu cầu

Bộ nhớ được phân thành các khung trang có nhiều loại kích thước.

### Question 5

0 / 1 pts

Hiện tượng các phần nhỏ không sử dụng trong bộ nhớ được tạo thành từ nhiều lần cấp phát và giải phóng vùng bộ nhớ gọi là hiện tượng gì?

Tráo đổi (Swaping)

Phân đoạn (Segment)

Phân trang (Paging)

#### Correct Answer

Phân mảnh (Defragment)

### Question 6

0 / 1 pts

Phân mảnh ngoại có thể giải quyết bằng biện pháp nào?

#### Correct Answer

Chia bộ nhớ chính thành các khung trang (frame).

Cấp phát liên tục bộ nhớ chính.

Sử dụng vùng nhớ ảo kích thước lớn.

Liên tục chống phân mảnh bộ nhớ chính.

### Question 7

0 / 1 pts

Không gian địa chỉ của một quá trình có kích thước 4GBytes. Số lượng bit cần dùng để đánh địa chỉ là bao nhiêu?

#### Correct Answers

32.0 (with margin: 0.0)

### Question 8

0 / 1 pts

Phân mảnh ngoại sẽ xảy ra khi giải thuật tìm lỗ trống nào sau đây được áp dụng?

First – fit

Worst – fit

Best – fit

#### Correct Answer

Phân mảnh ngoại luôn xảy ra, bất kể áp dụng giải thuật tìm lỗ trống nào.

### Question 9

0 / 1 pts

Cho bộ nhớ chứa các vùng nhớ không liên tục với độ lớn theo thứ tự sau: 10KB, 4KB, 20KB, 18KB, 7KB, 9KB, 13KB, và 15KB. Một tiến trình xin cấp phát 12 KB, và được nạp vào vùng nhớ 13KB. Giải thuật chọn lỗ trống trong bộ nhớ nào đã được áp

Swapping

**Correct Answer** Best – fit

Worst – fit

First – fit



**Question 10****0 / 2 pts**

Trong hệ thống quản lý bộ nhớ bằng phương pháp phân trang, hai tiến trình có thể liên lạc với nhau, dưới mô hình bộ nhớ chia sẻ được hiện thực bằng phương pháp nào?

**Correct Answer**

Sử dụng chung 1 khung trang trong bộ nhớ vật lý.

Đồng bộ liên tục 2 khung trang chứa 2 trang chia sẻ.

Nạp trang chia sẻ vào 2 khung trang của từng tiến trình.

2 tiến trình được bố trí sử dụng cùng 1 trang chia sẻ.

**Question 11****0 / 2 pts**

Trong kỹ thuật phân vùng nhớ kích thước cố định, độ đa lập trình của hệ thống là bao nhiêu?

là kích thước của bộ nhớ.

bằng với độ lợi CPU

**Correct Answer**

bằng số phân vùng đã chia.

không có giới hạn.

### Question 12

0 / 1 pts

Thông tin chứa trong Bảng phân trang là dùng để:

---

#### Correct Answer

Lưu thông tin vị trí nạp các trang của tiến trình trong bộ nhớ chính.

---

Lưu số trang của tiến trình đang hoạt động trong bộ nhớ chính.

---

Lưu số trang hợp lệ của tiến trình đã được cấp phát trước đó.

---

Cho biết thông tin các tiến trình có bao nhiêu trang.

### Question 13

0 / 2 pts

Hệ điều hành và các tiến trình cần được bảo vệ, tránh bị chỉnh sửa từ các tiến trình đang chạy, bằng cách nào?

---

tất cả tiến trình có địa chỉ nhớ luận lý hoàn toàn khác nhau.

tất cả các tiến trình nằm trong các không gian nhớ khác nhau.

tất cả tiến trình có địa chỉ nhớ luận lý hoàn toàn khác nhau.

---

#### Correct Answer

địa chỉ sinh ra từ CPU được kiểm tra có thuộc vùng nhớ hợp lệ hay không.

**Question 14****0 / 2 pts**

Chọn phát biểu sai về phương pháp quản lý bộ nhớ

Phân trang có thể dẫn đến hiện tượng phân mảnh nội.

**Correct Answer**

Phân hoạch tùy biến kích thước phân vùng (variable partitions) không bị phân mảnh ngoại.

Phân đoạn có thể dẫn đến hiện tượng phân mảnh ngoại.

Phân hoạch cố định kích thước phân vùng (fixed partitions) không bị phân mảnh ngoại.

**Unanswered Question 15****0 / 2 pts**

Lợi thế của phương pháp biên dịch tại thời điểm nạp (Load) là gì?

Có thể trao đổi (swap) tiến trình ra khỏi bộ nhớ chính và nạp lại vào một vị trí khác.

Làm tăng tốc độ xử lý tiến trình do vùng nhớ đang sử dụng sẽ có độ ưu tiên cao hơn

Giúp chương trình chạy hiệu quả hơn do không phải tính toán lại địa chỉ khi thực thi.

**Correct Answer**

Có thể nạp chương trình vào nhiều vị trí trong bộ nhớ mà không cần biên dịch lại.

### Question 16

0 / 2 pts

Phân mảnh nội xảy ra khi nào?

Hệ điều hành cấp phát liên tục bộ nhớ.

Khi hệ thống vận hành sau thời gian dài.

Khi các tiến trình được ghi/xóa liên tục

### Correct Answer

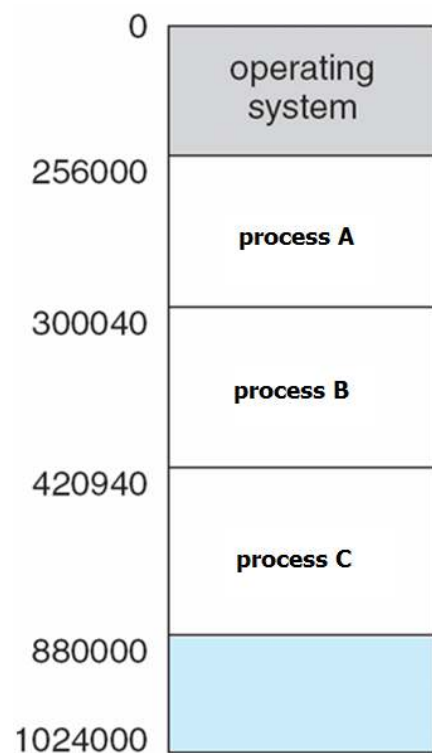
Hệ điều hành cấp phát bộ nhớ chính theo khối

### Question 17

0 / 4 pts

Cho mô hình cấp phát bộ nhớ liên tục như sau, trả lời các câu hỏi bên dưới bằng cách điền giá trị vào ô trống.

- Khi tiến trình B được xử lý trong CPU thì giá trị thanh ghi Base là .
- Khi tiến trình C được xử lý trong CPU thì giá trị thanh ghi Limit là .
- Khi tiến trình A được xử lý trong CPU, địa chỉ truy cập 300040 có hợp lệ không? (Điền Y hoặc N)
- Có bao nhiêu lỗ trống trong bộ nhớ?



---

**Answer 1**

**Correct Answer**

300040

---

**Answer 2:**

**Correct Answer**

459060

---

**Answer 3:**

**Correct Answer**

N

---

**Correct Answer**

n

---

**Answer 4:**

**Correct Answer**

1

---

**Question 18**

0 / 1 pts

Kích thước 1 trang (page size) trong hệ điều hành Windows là bao nhiêu bytes?

---

**Correct Answers**

4.0 (with margin: 0.0)

4096.0 (with margin: 0.0)

---

**Question 19**

0 / 1 pts

Hệ điều hành Windows sử dụng phương pháp cấp phát bộ nhớ nào?

---

Phân vùng.

**Correct Answer**

Phân trang.

Phân đoạn.

Liên tục.

---

**Question 20**

0 / 1 pts

Hệ điều hành ghi và cập nhật mỗi dòng trong bảng phân trang cho đối tượng nào sau đây?

mỗi câu lệnh

**Correct Answer**

mỗi tiến trình

mỗi khối lệnh

mỗi địa chỉ

### Question 21

0 / 1 pts

Với thanh ghi tái định vị (base) và thanh ghi giới hạn (limit), mỗi địa chỉ luận lý (logical address) phải có giá trị như thế nào với giá trị thanh ghi limit?

**Correct Answer**

@ < limit

@ = limit

@ <= limit

@ > limit

### Question 22

0 / 1 pts

Địa chỉ của bảng phân trang được lưu trữ ở đâu?

con trỏ stack của bảng phân trang

con trỏ trang

program counter

**Correct Answer**

con trỏ base của bảng phân trang

**Question 23**

0 / 3 pts

Cho bảng phân đoạn (Segment Table) như sau:

| Segment | Base | Limit |
|---------|------|-------|
| 0       | 219  | 600   |
| 1       | 2300 | 14    |
| 2       | 90   | 100   |
| 3       | 1327 | 580   |
| 4       | 1952 | 96    |

- Khi Segment 1 đang được xử lý trong CPU, địa chỉ luận lý 13 được dịch ra địa chỉ vật lý là bao nhiêu?

- Khi Segment 2 đang được xử lý trong CPU, địa chỉ vật lý lớn nhất hợp lệ là bao nhiêu?

- Kích thước của Segment 3 là bao nhiêu?

**Answer 1:**



---

**Correct Answer**

2313

---

**Answer 2:**

**Correct Answer**

189

---

**Answer 3:**

**Correct Answer**

580

---

**Question 24**

0 / 1 pts

Bộ nhớ ảo (Virtual Memory) là gì?

---

là một bản sao chép của bộ nhớ chính nhằm mục đích chia sẻ.

vùng nhớ mở rộng của bộ nhớ chính nhằm gia tăng kích thước bộ nhớ.

vùng nhớ chứa giá trị các biến số trong khi bộ nhớ chính chứa lệnh.

**Correct Answer**

vùng nhớ chứa những phần của tiến trình chưa được nạp vào bộ nhớ chính.

---

**Question 25**

0 / 1 pts

Phân trang theo yêu cầu (Demand Paging) hoạt động ra sao?

Tất cả mã nguồn của tiến trình cần được nạp và bộ nhớ khi bắt đầu chạy.

Chia tiến trình thành các khối nhớ kích thước tùy ý để tối ưu dung lượng.

Khi người dùng yêu cầu thì hệ thống mới tiến hành phân trang.

---

**Correct Answer**

Hệ thống có thể chỉ nạp những trang cần thiết vào khung trang.

---

**Question 26**

0 / 1 pts

Bất thường Belady nói đến số lỗi trang tăng khi cấp thêm khung trang xảy ra với các giải thuật nào?

LRU và OPT.

---

**Correct Answer**

Chỉ có FIFO.

Tất cả giải thuật.

LFU và LRU.

---

**Question 27**

0 / 2 pts

Trong hệ thống phân trang theo yêu cầu, nếu thời gian truy cập bộ nhớ chính là 200 nano-giây, thời gian xử lý lỗi trang là 8 mili-giây và một lỗi trang sẽ xảy ra trong 1000 lần truy vấn của CPU; thì thời gian truy cập hiệu quả là bao nhiêu?

EAT = 8 micro-giây.

EAT = 6.4 micro-giây.

**Correct Answer**

EAT = 8.2 micro-giây.

EAT = 8.2 mili-giây.

**Question 28**

0 / 2 pts

Copy-on-write (Sao chép khi ghi) là nguyên tắc gì?

Tiến trình cha được quyền ghi lên các trang chia sẻ trong khi tiến trình con chỉ được đọc.

Tiến trình con được cấp phát mới các trang trong bộ nhớ và nó sao chép lại từ tiến trình cha.

Tiến trình cha và con chia sẻ các trang trong bộ nhớ cho đến khi có thao tác cập nhật.  
(Correct Answer)

Bảo mật thông tin khi gửi và nhận thông điệp giữa các tiến trình.

**Question 29****0 / 2 pts**

Cho chuỗi tham khảo trang **7,0,1,2,0,3,0,4,2,3,0,3,0,3,2,1,2,0,1,7,0,1**. Hệ thống dùng 3 khung trang, với giải thuật FIFO (First in first out) thì số lỗi trang là bao nhiêu sau khi phục vụ hết chuỗi tham khảo trên?

**Correct Answers**

15.0 (with margin: 0.0)

**Question 30****0 / 2 pts**

Cho chuỗi tham khảo trang **7,0,1,2,0,3,0,4,2,3,0,3,0,3,2,1,2,0,1,7,0,1**. Hệ thống dùng 3 khung trang, với giải thuật LRU (Least Recently Used) thì số lỗi trang là bao nhiêu sau khi phục vụ hết chuỗi tham khảo trên?

**Correct Answers**

12.0 (with margin: 0.0)

**Question 31****0 / 2 pts**

Cho chuỗi tham khảo trang **7,0,1,2,0,3,0,4,2,3,0,3,0,3,2,1,2,0,1,7,0,1**. Hệ thống dùng 3 khung trang, với giải thuật Optimal thì số lỗi trang là bao nhiêu sau khi phục vụ hết chuỗi tham khảo trên?

**Correct Answers**

12.0 (with margin: 0.0)

**Question 32****0 / 3 pts**

Nguyên tắc thay thế trang địa phương (Local Replacement) là gì?

---

**Correct Answer**

Khi thay thế trang, trang nạn nhân được tìm trong phạm vi các khung đã cấp cho tiến trình đó.

---

Khi thay thế trang, trang nạn nhân được tìm trong phạm vi các khung đang ở gần con trỏ.

---

Khi thay thế trang, trang nạn nhân được tìm trong phạm vi các khung có trong bộ nhớ chính.

---

Khi thay thế trang, trang nạn nhân được tìm trong phạm vi các khung lân cận nhau.

---

**Question 33**

0 / 3 pts

Một hệ thống có 256 khung trang sẵn sàng và có 5 tiến trình vừa đến. Cho rằng kích thước tiến trình P1 là 10 trang, tổng kích thước các tiến trình còn lại là 50 trang. Hệ thống sẽ cấp cho P1 bao nhiêu khung trang nếu quy tắc “cấp phát tỉ lệ thuận” (Proportional Allocation) được áp dụng?

---

**Correct Answers**

10.0 (with margin: 0.0)

Chương 11+12

---

**Question 1**

0 / 1 pts

Thiết bị nào sau đây không phải là thiết bị lưu trữ thứ cấp?

**Correct Answer**

RAM

Magnetic tapes

Magnetic disks

USB Flash disk

Thuật toán định thời đĩa nào sẽ đáp ứng yêu cầu có khoảng di chuyển đầu đọc ngắn nhất?

**Correct Answer**

SSTF

FCFS

SCAN

LOOK

**Question 3**

0 / 1 pts

Thời gian để đầu đọc đĩa (trong HDD) di chuyển đến cylinder chứa sector cần đọc được gọi là gì?

---

**Correct Answer**

seek time

latency

arm time

sector time

**Question 4****0 / 1 pts**

Mạng máy tính nào sử dụng nhiều giao thức lưu trữ hơn là giao thức mạng?

---

**Correct Answer**

storage area network

local area network

wide area network

internet

**Question 5****0 / 1 pts**

Đĩa cứng chứa phân vùng khởi động được gọi là gì?

---

boot disk **Correct Answer**

end disk

start-up disk

hard disk

### Question 6

0 / 1 pts

Những sector không còn dùng được trên đĩa cứng thường được gọi là gì?

bad blocks   **Correct Answer**  
good blocks

destroyed blocks

unusable blocks

### Question 7

0 / 1 pts

Trong phần lớn các hệ thống máy tính, bootstrap chứa ở đâu?

ROM   **Correct Answer**  
RAM

Cache

HDD

### Question 8 Thiết bị nào lưu trữ thông tin bằng từ tính trên một mặt đĩa

**Correct Answer**  
HDD

HDD và CD/DVD

HDD và SSD

RAM và HDD



### Question 9

0 / 1 pts

Tất cả đầu đọc của một đĩa cứng được gắn lên thiết bị nào sau đây để cùng di chuyển với nhau?

---

#### Correct Answer

disk arm

---

spindle

---

track

---

cylinder

### Question 10 :Đặc trưng của RAID 5 là gì?

#### Correct Answer

Khối parity phân bố trên mọi đĩa cứng.

---

Không sử dụng parity, ghi bản sao lên 2 đĩa.

Khối parity nằm trên cùng một đĩa cứng.

Sử dụng 2 khối parity độc lập

**Question 11****0 / 2 pts**

Đặc trưng của RAID 6 là gì?

**Correct Answer**

Sử dụng 2 khối parity độc lập.

Chia dữ liệu ra ghi trên nhiều đĩa cứng.

Khối parity phân bố trên mọi đĩa cứng.

Sao y đĩa cứng.

**Question 12 : RAID nào sau đây có sử dụng bit kiểm tra parity để bảo vệ dữ liệu?****Correct Answer**

RAID 4

RAID 1+0

RAID 0

RAID 1

**Question 13****0 / 2 pts**

Quá trình thay thế khối đĩa bị hỏng (bad block) không thể hoàn toàn tự động bởi vì sao?

**Correct Answer**

dữ liệu trong khối đĩa hỏng thường bị mất.

dữ liệu trong khối đĩa hỏng không thể thay thế.

khối đĩa hỏng không chứa dữ liệu.

đĩa cứng không hoạt động nếu có khối đĩa hỏng.

**Question 14: RAID mức 1 có đặc điểm gì?****Correct Answer**

sao y các đĩa cứng với nhau.

chia dữ liệu để ghi trên nhiều đĩa cứng.

sao lưu dữ liệu sau một khoảng thời gian.

tăng tốc độ đọc/ghi của đĩa cứng.

**Question 15****0 / 2 pts**

Số lượng đĩa cứng tối thiểu để áp dụng RAID1 là bao nhiêu?

---

**Correct Answer**

2

---

1

---

4

---

5

**Question 16:** Trong các RAID sau đây, RAID nào cung cấp nhiều nhất dung lượng khả dụng cho người dùng?

---

**Correct Answer**

RAID 0

---

RAID 1

---

RAID 5

---

RAID 6

**Question 17:**Chương trình nào khởi tạo các đặc tính của hệ thống (như là các thanh ghi CPU, các điều khiển thiết bị và nội dung bộ nhớ chính), rồi khởi động hệ điều hành?

**Correct Answer**

bootstrap

---

bootloader

---

main

---

ROM

**Question 18:**Thời gian mean time để bị lỗi của một đĩa cứng là 200.000 giờ. Nếu một mảng sử dụng 5 đĩa cứng như vậy thì mean time to failure (mttf) của mảng này là bao nhiêu giờ?

**Correct Answer**

40.000 giờ

---

200.000 giờ

---

1.000.000 giờ

---

Không đủ thông tin để kết luận.

### Question 19

0 / 2 pts

Một mảng nhiều đĩa cứng sẽ dễ xảy ra nhiều hỏng hóc hơn là một đĩa cứng độc lập. Vậy cấu trúc RAID có thể bảo vệ dữ liệu tốt hơn một đĩa cứng độc lập bằng cách nào?

#### Correct Answer

Áp dụng sao y đĩa cứng và khối parity.

Chia dữ liệu ra ghi trên nhiều đĩa cứng.

Sử dụng các đĩa cứng chất lượng hơn.

Sử dụng phần cứng chuyên dụng

**Question 20:** Hai bước mà hệ điều hành thực hiện để một đĩa cứng có thể bắt đầu sử dụng là gì?

#### Correct Answer

Phân vùng đĩa, sau đó định dạng luận lý.

Tạo không gian hoán chuyển và vùng đệm.

Tạo vùng đệm, sau đó định dạng luận lý.

Định dạng luận lý, sau đó tạo vùng đệm.

**Question 21:** Thông tin cung cấp cho bộ điều khiển đĩa chứa trong một sector là những thông tin gì?

**Correct Answer**

error correcting codes (ECC) & sector number

---

ain section & disk identifier

---

sector number & main section

---

disk identifier & sector number

**Question 22:** Thời gian để mặt đĩa quay để đầu đọc đến được sector mong muốn đọc/ghi được gọi là gì?

**Correct Answer**

rotational latency

---

random access time

---

seek time

---

positioning time

### Question 23

0 / 2 pts

Băng thông đĩa cứng là gì?

#### Correct Answer

tổng số byte được truyền sẽ chia cho tổng thời gian giữa yêu cầu dịch vụ đầu tiên và hoàn thành lần chuyển cuối cùng

tổng thời gian giữa yêu cầu dịch vụ đầu tiên và hoàn thành lần chuyển cuối cùng

tổng số byte được truyền

là tốc độ đọc hay ghi của đĩa cứng tùy vào việc đọc nhanh hơn hay ghi nhanh hơn.

### Question 24

0 / 2 pts

Mỗi khi một tiến trình cần đọc hay ghi với một đĩa cứng, nó cần phải làm gì?

#### Correct Answer

Gửi lời gọi hệ thống đến hệ điều hành.

Gửi lời gọi hệ thống đến CPU.

Gửi yêu cầu đến đĩa cứng.

Tạo một liên kết đến đĩa cứng và bắt đầu đọc/ghi.



**Question 25:** Hệ thống nào sau đây có thể áp dụng RAID 05?

**Correct Answer**

3 nhóm đĩa, mỗi nhóm có 2 đĩa cứng.

2 nhóm đĩa, mỗi nhóm có 3 đĩa cứng.

1 nhóm đĩa và có 5 đĩa cứng.

4 nhóm đĩa, mỗi nhóm có 1 đĩa cứng.

**Question 26:** Cho hàng chờ đĩa với các yêu cầu I/O trên các cylinder sau đây:

98 183 37 122 14 124 65 67

Giải thuật định thời đĩa FCFS (first come first serve) được áp dụng, tổng số cylinder mà đầu đọc sẽ di chuyển qua là bao nhiêu, giả sử rằng đầu đọc ban đầu nằm ở cylinder 53.

**Correct Answers**

640.0 (with margin: 0.0)

**Question 27:** Cho hàng chờ đĩa với các yêu cầu I/O trên các cylinder sau đây:

98 183 37 122 14 124 65 67

Giải thuật định thời đĩa SSTF (shortest seek time first) được áp dụng, tổng số cylinder mà đầu đọc sẽ di chuyển qua là bao nhiêu, giả sử rằng đầu đọc ban đầu nằm ở cylinder 53.

**Correct Answers**

236.0 (with margin: 0.0)

**Question 28****0 / 3 pts**

Đầu đọc bắt đầu từ một hướng, đáp ứng các yêu cầu I/O mà nó bắt gặp trên đường di chuyển, khi đến vành đĩa (hoặc tâm đĩa), đầu đọc đảo chiều di chuyển rồi tiếp tục đáp ứng các yêu cầu I/O. Đó là mô tả của giải thuật định thời đĩa nào?

**Correct Answer**

SCAN

LOOK

C-LOOK

C-SCAN

**Question 29****0 / 3 pts**

Đầu đọc bắt đầu từ một hướng, đáp ứng các yêu cầu I/O mà nó bắt gặp trên đường di chuyển, khi đến vành đĩa (hoặc tâm đĩa), đầu đọc di chuyển trở về phía bên kia mà không đáp ứng các yêu cầu I/O trong quá trình này. Đó là mô tả của giải thuật định thời đĩa nào?

**Correct Answer**

C-SCAN

SCAN

LOOK

---

C-LOOK

**Question 30**

**0 / 3 pts**

Đầu đọc di chuyển qua lại, mỗi chiều di chuyển đầu đọc đáp ứng đến yêu cầu ở cylinder xa nhất rồi đảo chiều, mà không cần phải di chuyển đến tâm đĩa hay vành đĩa. Đó là mô tả của giải thuật định thời đĩa nào?

---

**Correct Answer**

LOOK

---

SCAN

---

C-LOOK

---

C-SCAN

### Question 31

0 / 1 pts

Nếu một vài thiết bị sử dụng các dây dẫn để truyền nhận dữ liệu với nhau bên trong một máy tính, kết nối đó được gọi là gì?

#### Correct Answer

Bus

Monitor

Wirefull

CPU

**Question 32:** Khi một thiết bị A có sử dụng cáp liên lạc với thiết bị B, và thiết bị B cũng có cáp để liên lạc với thiết bị C, thiết bị C được cắm vào một cổng phù hợp trên máy tính. Sự sắp xếp vừa nêu được gọi là?

#### Correct Answer

daisy chain

port

bus

cable

**Question 33****0 / 1 pts**

Cơ chế phần cứng cho phép một thiết bị gửi tín hiệu báo đến CPU được gọi là gì?

**Correct Answer**

interrupt

polling

driver

controlling

**Question 34****0 / 1 pts**

Tập hợp các đường dây dẫn và các giao thức gửi nhận thông điệp trên các đường dây đó được gọi là gì?

**Correct Answer**

bus

port

node

IPC

### Question 35

0 / 1 pts

Phần cứng I/O bao gồm những gì?

#### Correct Answer

Bus, Controller, cổng I/O và các thanh ghi.

CPU, thiết bị I/O, dây dẫn kết nối.

Thiết bị I/O, controller và các giao thức truyền nhận.

Thiết bị I/O, bus, cổng I/O

### Question 36

0 / 1 pts

Tập hợp các thiết bị vận hành cổng giao tiếp, bus và thiết bị được gọi là gì?

#### Correct Answer

controller

driver

host

bus

**Question 37****0 / 2 pts**

Cung cấp một giao diện truy xuất thiết bị đồng nhất tới hệ thống nhập xuất bên dưới là trách nhiệm của bộ phận nào?

**Correct Answer**

Driver của thiết bị

Thiết bị

Bus

Hệ thống nhập xuất

**Question 38****0 / 2 pts**

Một cổng I/O thông thường có 4 thanh ghi, bao gồm thanh ghi trạng thái (status), thanh ghi điều khiển (control) và 2 thanh ghi nào sau đây?

**Correct Answer**

data in, data out

system in, system out

flow in, flow out

input, output

**Question 39****0 / 2 pts**

Thanh ghi nào được máy tính ghi để gửi dữ liệu ra?

---

**Correct Answer**

data out

status

control

data in

**Question 40****0 / 2 pts**

Tín hiệu mà CPU gửi ra sau khi hoàn thành xử lý một câu lệnh (chỉ thị) được gọi là gì?

---

**Correct Answer**

interrupt request line

interrupt bus

interrupt receive line

interrupt sense line



**Question 41****0 / 2 pts**

Bộ phận nào đảm nhiệm vai trò phát hiện lý do của một ngắt, thực thi các thao tác cần thiết và gửi kết quả đến CPU để CPU phản hồi ngắt vừa xảy ra?

**Correct Answer**

interrupt handler

device driver

interrupt request line

mother board

**Question 42: Có thể phân loại đường dây yêu cầu ngắt như thế nào?****Correct Answer**

Ngắt mặt nạ hay không (maskable hay nonmaskable interrupts)

Ngắt bị khóa hay không (blocked hay nonmaskable interrupts)

Ngắt mặt nạ và ngắt bị khóa (maskable hay blocked interrupts)

Ngắt hệ thống và ngắt người dùng (system hay user interrupts)

### Question 43

0 / 2 pts

Loại ngắt nào được dùng cho những sự kiện như là lỗi “bộ nhớ bị lỗi không thể hồi phục”.

---

#### Correct Answer

Ngắt không có mặt nạ (nonmaskable interrupts)

Ngắt bị khóa (blocked interrupts)

Ngắt có mặt nạ (maskable interrupts)

Ngắt hệ thống (system interrupts)

### Question 44

0 / 2 pts

Thanh ghi “data-in” của cổng I/O được sử dụng làm gì?

---

#### Correct Answer

Đọc bởi máy tính để lấy dữ liệu vào.

Đọc bởi bộ điều khiển để lấy dữ liệu vào.

Được máy tính ghi dữ liệu ra.

Được máy tính gửi lệnh khởi động.

**Question 45****0 / 2 pts**

Bit nào được máy tính thiết lập khi một lệnh cần được thực thi bộ bộ điều khiển nhập xuất?

---

**Correct Answer**

command-ready

status

write

Control

**Question 46****0 / 3 pts**

Một phần cứng được truy xuất thông qua việc đọc và ghi trên vùng bộ nhớ cụ thể cấp trước là mô tả của kỹ thuật nào?

---

**Correct Answer**

memory-mapped I/O

controller-mapped I/O

bus-mapped I/O

port-mapped I/O

### Question 47

0 / 3 pts

Loại thiết bị nào thì cần driver để hệ điều hành có thể giao tiếp với chúng?

#### Correct Answer

Mọi loại thiết bị.

Chỉ có thiết bị kiểu block.

Chỉ có thiết bị kiểu network.

Chỉ có thiết bị kiểu character.

### Question 48

0 / 3 pts

Phần cứng nào kích hoạt một số thao tác sau các lệnh nhất định hoàn tất?

programmable interval timer **Correct Answer**

interrupt timer

programmable timer

CPU timer

### Question 49

0 / 3 pts

Bộ đệm (buffer) nào sau đây lưu trữ tạm thời dữ liệu ra cho một thiết bị?

Spool **Correct Answer**

output

status

magic

---