

# Hướng dẫn giải chi tiết đề ôn cuối kỳ Hệ điều hành

**Lưu ý:** Tài liệu chỉ mang tính tham khảo, không phải là lời giải đúng.

## Câu 1.

Giả sử mỗi quá trình được cấp phát không gian nhớ có kích thước 32768 bytes. Một quá trình A yêu cầu được cấp phát vùng nhớ như sau: text: 10284 bytes, data: 9786 bytes, stack: 11770 bytes.

- Hãy tính số trang nhớ được cấp tối đa cho mỗi quá trình nếu kích thước mỗi trang là 4096 bytes.
- Hệ thống có thể đáp ứng toàn bộ yêu cầu của quá trình A tại cùng một thời điểm không nếu kích thước mỗi trang là 4096 bytes ?
- Nếu kích thước mỗi trang nhớ là 512 bytes, thì số trang nhớ cần cấp phát cho stack là ?
- Hệ thống có thể đáp ứng toàn bộ yêu cầu của quá trình tại cùng một thời điểm không nếu kích thước mỗi trang là 512 bytes ?

## Solution.

a) Số trang nhớ có thể cấp phát =  $\lfloor \frac{\text{memory space}}{\text{page size}} \rfloor = \frac{32768}{4096} = 8$

b) Số trang nhớ cần để đáp ứng toàn bộ quá trình A =  $\sum \lceil \frac{\text{memory request}}{\text{page size}} \rceil = \lceil \frac{10284}{4096} \rceil + \lceil \frac{9786}{4096} \rceil + \lceil \frac{11770}{4096} \rceil$   
 $= 3 + 3 + 3 = 9$

Được cấp tối đa 8 trang  $\Rightarrow$  không đáp ứng được.

c) Số trang cần cấp phát =  $\lceil \frac{\text{stack request}}{\text{page size}} \rceil = \lceil \frac{11770}{512} \rceil = 23$

d) Tương tự câu b, Pages request = 64 = Max pages  $\Rightarrow$  đáp ứng được. □

## Câu 2.

Một partition đĩa có kích thước 256M được format với hệ thống tập tin FAT. Mỗi block đĩa chiếm 1M.

- Tính kích thước một entry của bảng FAT và từ đó tính kích thước của bảng FAT.
- Cho thông tin của một tập tin như bên dưới. Hãy xác định các block đĩa chứa nội dung của tập tin.

File name	1st block	Size
total.txt	7	7MB

Biết bảng FAT có nội dung như sau:

0	8
1	3
2	6
3	2
4	1
5	end-of-file
6	5
7	4
8	7
...	...

### Solution.

a)  $Kích\ thước\ 1\ entry = \log_2 \frac{disk\ size}{block\ size} = \log_2 \frac{256M}{1M} = 8\ bits = 1B$   
 $\Rightarrow$   $Kích\ thước\ bảng\ FAT = 256B$

b) 1, 2, 3, 4, 5, 6, 7. □

### Câu 3.

Cho biết một block đĩa chiếm 4KB, mỗi entry chiếm 4B. Giả sử một i-node có thể chứa 12 pointer trở tới các data block, 1 pointer trở tới indirect block, 1 pointer trở tới double indirect block, và 1 pointer trở tới 1 triple indirect block.

- Hãy tính kích thước tối đa mà một tập tin có thể có.
- Xác định số con trỏ của i-node cần sử dụng cho tập tin toto.txt ở câu 2.
- Bao nhiêu free block được sử dụng để cấp phát cho tập tin có kích thước tối đa ở câu a.

### Solution.

a)  $Max\ size = \sum data\ block = 12 \cdot 4KB + \frac{4KB}{4B} \cdot 4KB + \frac{4KB}{4B} \cdot \frac{4KB}{4B} \cdot 4KB + \frac{4KB}{4B} \cdot \frac{4KB}{4B} \cdot \frac{4KB}{4B} \cdot 4KB$   
 $= 48KB + 4MB + 4GB + 4TB$

b) Các pointer được sử dụng tính theo khả năng lưu trữ dữ liệu từ ít đến nhiều, đến khi đủ để chứa file cần thiết.

Pointers	Size of file
0	0
1	$0 < Size \leq 4KB$
...	...
12	$44KB < Size \leq 48KB$
13	$48KB < Size \leq 4MB + 48KB$
14	$4MB + 48KB < Size \leq 4GB + 4MB + 48KB$
15	$4GB + 4MB + 48KB < Size \leq 4TB + 4GB + 4MB + 48KB$

Vì file toto.txt có kích thước 7MB nên cần 14 pointers.

c)  $Số\ free\ blocks = 12 + \frac{4KB}{4B} + \frac{4KB}{4B} \cdot \frac{4KB}{4B} + \frac{4KB}{4B} \cdot \frac{4KB}{4B} \cdot \frac{4KB}{4B} = 12 + 2^{10} + 2^{20} + 2^{30} = 1074791436$  □

### Câu 4.

Cho chuỗi tham chiếu trang nhớ dưới đây:

0 2 1 3 0 1 4 0 1 2 3 4

- a) Hãy tính số lỗi trang khi sử dụng giải thuật Clock với 3 frame trống.  
b) Ước lượng giá trị Hit ratio tương ứng với hai giải thuật.

### Solution.

a)

0*	0*	→ 0*	3*	3*	3*	→ 3	→ 3	1*	1	→ 1	4*
→	2*	2*	→ 2	0*	0*	0	0*	→ 0*	2*	2*	→ 2*
	→	1*	1	→ 1	→ 1*	4*	4*	4*	→ 4	3*	3*
<i>F</i>	<i>F</i>	<i>F</i>	<i>F</i>	<i>F</i>		<i>F</i>		<i>F</i>	<i>F</i>	<i>F</i>	<i>F</i>

Số lỗi trang: 10

- b) Giải thuật 1:  $Hit\ ratio = \frac{2}{12} = \frac{1}{6}$  ; Giải thuật 2:  $Hit\ ratio = \frac{2}{12} = \frac{1}{6}$

□

### Câu 5.

Cho bảng dữ liệu dưới đây:

Virtual page	Valid	Physical page
0	No	
1	No	
2	Yes	1
3	No	
4	Yes	3
5	No	
6	Yes	0
7	Yes	2

Kích thước mỗi trang nhớ là 2K. Một quá trình được cấp phát 4 frame.

- a) Hãy xác định khoảng địa chỉ luận lý cho mỗi trang.  
b) Hãy xác định các địa chỉ luận lý có thể gây lỗi trang.  
c) Xác định địa chỉ vật lý cho các địa chỉ luận lý sau: 8500, 14000, 5000, 2100.

### Solution.

□

Virtual page	Logical address space	Error	Logical address	Physical address
0	0x0000 → 0x07FF	Yes	2100	Invalid
1	0x0800 → 0x0FFF	Yes	x	Invalid
2	0x1000 → 0x17FF	No	5000	2952
3	0x1800 → 0x1FFF	Yes		
4	0x2000 → 0x27FF	No	8500	6452
5	0x2800 → 0x2FFF	Yes		
6	0x3000 → 0x37FF	No	14000	1712
7	0x3800 → 0x3FFF	No	$k$	$k - (Virtual\ page - Physical\ page) \cdot Page\ size$

### Câu 6.

Cho hai quá trình  $P_0$  và  $P_1$ . Biến *turn* được khởi tạo giá trị 0. Đoạn mã của  $P_0$  được trình bày như sau:

```
...
while (turn != 0) { } /* Do nothing and wait. */
Critical Section /* . . . */
turn = 0;
...
```

Với  $P_1$ , thay 0 bởi 1 với đoạn code trên. Giải pháp trên có thỏa mãn bài toán loại trừ tương hỗ không ?

### Solution.

*Không, vì không thỏa mãn Progress cho entry section: Quá trình không thể vào CS bằng  $P_0$  dù đến đầu tiên (vào  $P_1$ ) và ngược lại.* □

### Câu 7.

Cho 5 quá trình và các thông tin như sau:

Process	Burst Time	Priority	Arrival time
P1	10	3	0
P2	1	1	1
P3	3	3	3
P4	1	4 (highest)	5
P5	5	2	6

a) Hãy vẽ giản đồ Gantt biểu diễn sự thực thi của các quá trình trên với các giải thuật định thời preemptive Priority, Round Robin ( $q=2$ ), SRTF, SJF.

b) Hãy tính thời gian quay vòng trung bình của các quá trình tương ứng với các giải thuật định thời trên.

### Solution.

a)

		Preemptive Priority																				
CPU		X	P1	P1		P1		P4	P1					P3			P5					P2
Time		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Arrival time		P1		P2		P3		P4		P5												
Queue		P1	P1	P1		P4		P1					P3	P5								P2
		P2		P3		P1		P3					P5	P2								
		P2			P3		P5					P2										
		P2					P2															



		Round Robin																				
CPU	X	P1		P2	P1		P3		P1		P4	P5		P3	P1		P5		P1		P5	
Time	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
Arrival time	P1	P2		P3		P4		P5														
Queue	P1	P2	P2	P1		P3		P1	P1	P4		P5		P3		P1	P5		P1		P5	
		P1		P3		P1		P4	P4	P5		P3		P1		P5	P1		P5			
						P4		P5	P5	P3		P1		P5								
								P3		P1												

		SRTF																			
CPU	X	P1	P2	P1	P3		P3	P4	P5					P1							
Time	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Arrival time	P1	P2		P3		P4		P5													
Queue	P1	P2	P1	P3		P3		P4	P5		P1										
		P1		P1		P4		P5	P1												
						P1		P1													

		SJF																				
CPU	X	P1										P2	P4	P3			P5					
Time	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
Arrival time	P1	P2		P3		P4		P5														
Queue	P1	P2		P2		P2		P2		P2					P4	P3	P5					
				P3		P3		P3		P4					P3	P5						
						P4		P4		P3					P5							
								P5		P5												

b)

$$PP: ATT = \frac{(11-0)+(20-1)+(14-3)+(6-5)+(19-6)}{5} = 11s$$

$$RR: ATT = \frac{(19-0)+(3-1)+(13-3)+(10-5)+(20-6)}{5} = 10s$$

$$SRTF: ATT = \frac{(20-0)+(2-1)+(6-3)+(7-5)+(12-6)}{5} = 6,4s$$

$$SJF: ATT = \frac{(10-0)+(11-1)+(15-3)+(12-5)+(20-6)}{5} = 10,6s$$

□

### Câu 8.

Cho các quá trình như bảng dưới đây:

Process	Arrival Time	Burst Time (ms)
P1	0.0	8
P2	0.4	4
P3	1.0	1

Giả thiết CPU sẽ ở trạng thái idle ở 1ms đầu tiên, sau đó giải thuật SJF được sử dụng để định thời các quá trình.

### Solution.

CPU	X		P3		P2				P1							
Time	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	

□

### Câu 9.

Có 5 quá trình ở hàng đợi ready. Thời gian CPU-burst của các quá trình lần lượt như sau: 9,6,3,5,X. Các quá trình nên được thực hiện theo thứ tự nào để giảm thời gian đáp ứng trung bình?

### Solution.

*Cần giảm thời gian đáp ứng trung bình nên ta sử dụng giải thuật SJF để sắp xếp thứ tự:*

□

X's Burst time	Arrangement
$0 < X \leq 3$	X,3,5,6,9
$3 < X \leq 5$	3,X,5,6,9
$5 < X \leq 6$	3,5,X,6,9
$6 < X \leq 9$	3,5,6,X,9
$9 < X < \infty$	3,5,6,9,X

### Câu 10.

Thuận lợi và bất lợi của việc hiện thực thread trong user space?

#### Solution.

*Thuận lợi:*

- Khởi tạo và quản lý thread nhanh hơn kernel.
- Tăng khả năng đáp ứng.
- Chia sẻ được dữ liệu.
- Tiết kiệm chi phí hệ thống.
- Tận dụng được đa xử lý.

*Bất lợi:*

- Khi một thread bị block thì các thread khác của process không tiến triển được.

□

### Câu 11.

Giải pháp Peterson có thể vận dụng cho bài toán đồng bộ khi định thời theo chế độ quyết định preemptive không? Nếu theo chế độ quyết định non-preemptive thì sẽ như thế nào ?

#### Solution.

*Không được, vì sẽ xảy ra tranh chấp vùng CS nếu có 2 process cùng vào  $P_0$  hoặc  $P_1$ . Nếu theo chế độ non-preemptive thì giải pháp Peterson vẫn đúng.*

□

### Câu 12.

Có lỗi xảy ra không khi thanh ghi base và thanh ghi limit có cùng giá trị ?

#### Solution.

*Không, vì giá trị trên 2 thanh ghi này không liên quan gì đến nhau.*

□