

Bài 1: Kích thước khung bộ nhớ là 4096 bytes. Hãy chuyển địa chỉ logic 8207, 4300 sang địa chỉ vật lý biết rằng bảng trang như sau:

Số trang	Số khung
0	13
1	15
2	10
3	3
4	22
5	7

Kích thước khung bộ nhớ 4096 byte = 2^{12}

Địa chỉ logic 8207:

$$p = 8207 / 4096 = 2 \quad \text{Số thứ tự trang} = 2$$

Theo bảng trang, trang 2 nằm ở số khung trang 10.

$$o = 8207 \% 4096 = 15$$

Do vậy, địa chỉ vật lý tương ứng là $10 * 4096 + 15 = 409675$.

Địa chỉ logic 4300:

$$p = 4300 / 4096 = 1 \quad \text{Số thứ tự trang} = 1$$

Theo bảng trang, trang 1 nằm ở số khung trang 15.

$$o = 4300 \% 4096 = 204$$

Do vậy, địa chỉ vật lý tương ứng là $15 * 4096 + 204 = 61644$.

Bài 2: Không gian địa chỉ logic của tiến trình gồm 17 trang, mỗi trang có kích thước 2048B được ánh xạ vào bộ nhớ vật lý có 27 khung

a. Để biểu diễn địa chỉ logic cần tối thiểu bao nhiêu bit

b. Để biểu diễn địa chỉ vật lý cần bao nhiêu bit

Kích thước trang là 2048B = 2^{11} => cần 11 bit để biểu diễn

Địa chỉ logic có 17 trang ($2^4 < 17 < 2^5$) => cần 5 bit để biểu diễn

⇒ Cần 11+5 = 16 bit để biểu diễn địa chỉ logic và địa chỉ vật lý

Bộ nhớ vật lý có 27 khung ($2^4 < 27 < 2^5$) => cần 5 bit để biểu diễn

⇒ Cần 11+5 = 16 bit để biểu diễn địa chỉ logic và địa chỉ vật lý

Câu 4/Đề 01 Giả sử không gian nhớ logic gồm 8 trang, mỗi trang có kích thước 4096B, bộ nhớ vật lý gồm 64 khung. Bảng trang được cho dưới đây:

0	13
1	10
2	4
3	23
4	
5	17
6	28
7	31

Để biểu diễn địa chỉ vật lý và địa chỉ logic trong trường hợp này cần bao nhiêu bit?

Tính địa chỉ vật lý cho những địa chỉ logic sau: 32,1190, 4700, 13046, 20580.

a) Kích thước trang là $4096 = 2^{12}$

độ dịch o là 12 bit

Bộ nhớ vật lý 64 khung = 2^6 cần 6 bit để biểu diễn

Do đó cần $12 + 6 = 18$ bit để biểu diễn địa chỉ vật lý.

Không gian nhớ logic = $8.4096 = 2^{15}$ (Đây là cách 2 rất ngắn >:))

Do đó cần 15 bit để biểu diễn địa chỉ logic.

b) Địa chỉ logic 32

$p = 32 / 4096 = 0$ Số thứ tự trang là 0 số khung trang là 13

$o = 32 \bmod 4096 = 32$

Khi đó địa chỉ vật lý = $13 \cdot 4096 + 32 = 53280$

Địa chỉ logic 1190: = 54438

Địa chỉ logic 4700: = 41564

Địa chỉ logic 13046: = 94966

Địa chỉ logic 20580: = 69732.

Bài 3: Bộ nhớ vật lý có 4 khung. Thứ tự truy cập các trang là 1, 2, 3, 4, 2, 1, 5, 6, 2, 1, 2, 3, 7, 6, 3, 2, 1, 2, 3, 6. Vẽ sơ đồ cấp phát bộ nhớ và Có bao nhiêu sự kiện thiếu trang xảy ra nếu sử dụng:

- Thuật toán tối ưu

- FIFO

- LRU

- Đồng hồ

Thuật toán tối ưu: Dự đoán tương lai.

1	2	3	4	2	1	5	6	2	1	2	3	7	6	3	2	1	2	3	6
1	1	1	1	1	1	1	1	1	1	1	1	7	7	7	7	1	1	1	1
	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
		3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
			4	4	4	5	6	6	6	6	6	6	6	6	6	6	6	6	6

⇒ Có 4 sự kiện đổi trang.

FIFO: Vào trước ra trước

1	2	3	4	2	1	5	6	2	1	2	3	7	6	3	2	1	2	3	6
1	1	1	1	1	1	5	5	5	5	5	3	3	3	3	3	1	1	1	1
	2	2	2	2	2	2	6	6	6	6	6	7	7	7	7	7	7	3	3
		3	3	3	3	3	3	2	2	2	2	2	6	6	6	6	6	6	6
			4	4	4	4	4	4	1	1	1	1	1	1	2	2	2	2	2

⇒ Có 10 sự kiện đổi trang.

LRU: Tương tự như FIFO nhưng quan tâm thêm lần truy cập cuối cùng của trang.

1	2	3	4	2	1	5	6	2	1	2	3	7	6	3	2	1	2	3	6
1	1	1	1	1	1	1	1	1	1	1	1	1	6	6	6	6	6	6	6
	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
		3	3	3	3	5	5	5	5	5	3	3	3	3	3	3	3	3	3
			4	4	4	4	6	6	6	6	6	7	7	7	7	1	1	1	1

⇒ Có 6 sự kiện đổi trang.

Thuật toán đồng hồ: Dựa trên FIFO thêm thông tin * và

Khi có nhu cầu đổi trang, hệ điều hành kiểm tra trang đang bị trở tới. Nếu bit U của trang bằng 0, trang sẽ bị đổi ngay. Ngược lại, nếu bit U bằng 1, hệ điều hành sẽ đặt bit U bằng 0, chuyển sang trang tiếp theo trong danh sách và lặp lại thủ tục trên.

1	2	3	4	2	1	5	6	2	1	2	3	7	6	3	2	1	2	3	6
1*	1*	1*	-1*	-1*	-1*	5*	5*	5*	-5*	-5*	3*	3*	3*	3*	-3*	1*	1*	1*	1*
-	2*	2*	2*	2*	2*	-2	6*	6*	6*	6*	-6	7*	7*	7*	7*	-7	-7	3*	3*

	-	3*	3	3*	3*	3	-3	2*	2*	2*	2	-2	6*	6*	6*	6	6	-6	-6*
		-	4*	4*	4*	4	4	-4	1*	1*	1	1	-1	-1	2*	2	2*	2*	2*

⇒ Có 10 sự kiện đối xứng.

Câu 4/Đề 04

3	2	1	4	5	2	3	1	6	7	5	6	8	9	7	4	2	5
3*	3*	3*	-3*	5*	5*	5*	-5*	5	7*	7*	7*	-7*	9*	9*	9*	-9*	5*
-	2*	2*	2*	-2	-2*	2	2	6*	-6*	6	6*	6*	-6	7*	7*	7*	-7
	-	1*	1*	1	1	3*	3*	-3*	3	5*	5*	5*	5	-5	4*	4*	4
		-	4*	4	4	-4	1*	1*	1	-1	-1	8*	8	8	-8	2*	2
				F		F	F	F	F	F		F	F	F	F	F	F

Bài 4: Bộ nhớ có kích thước 1MB. Sử dụng phương pháp kẻ cận (buddy system) để cấp phát cho các tiến trình lần lượt với kích thước như sau: A: 112KB, B: 200KB, C: 150KB, D: 50KB

1MB				
A=112KB	128KB	256KB	512KB	
A=112KB	128KB	B=200KB	512KB	
A=112KB	128KB	B=200KB	C=150KB	256KB
A=112K	D=50KB	B=200KB	C=150KB	256KB

Cấp A

Cấp B

Cấp C

Cấp D

Bài 5: Cho danh sách các tiến trình mà HĐH phải lập lịch hoạt động với thông tin như sau

Tên tiến trình	Thời gian đến (Arrival time)	Thời gian chạy (Burst Time)
P1	2	12
P2	0	11
P3	2	7
P4	3	15
P5	4	8

Tiến hành lập lịch hoạt động của các tiến trình trên với các thuật toán sau:

1. FCFS
2. SPF
3. SRTF

4. Round robin (t = 3)

Với mỗi thuật toán, so sánh các thông số sau:

- Thời gian chờ đợi trung bình

1. FCFS

Tiến trình	P2	P1	P3	P4	P5
Thời gian chạy	11	12	7	15	8
Thời gian chờ	0	9	21	27	41

Thời gian chờ trung bình: $(9 + 21 + 27 + 41)/5 = 98/5$

2. SPF : Ưu tiên tiến trình ngắn nhất

Tiến trình	P2	P3	P5	P1	P4
Thời gian chạy	11	7	8	12	15
Thời gian chờ	0	9	14	24	35

Thời gian chờ trung bình: $(9 + 14 + 24 + 35)/5 = 82/5$

3. SRTF : Ưu tiên thời gian còn lại ngắn nhất

Tiến trình	P2	P3	P5	P2	P1	P4
Thời gian chạy	2	7	8	9	12	15
Thời gian chờ	0	0	5	15	24	35

Thời gian chờ trung bình: $(5 + 15 + 24 + 35)/5 = 79/5$

4. RR (t=3)

Câu 4 đề 3

Tiến trình	Thời điểm xuất hiện	Độ dài
P1	0	8
P2	2	4
P3	0	2
P4	4	5

Giấy thứ	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Tiến trình	P1	P1	P3	P3	P1	P1	P2	P2	P4	P4	P1	P1	P2	P2	P4	P4	P1	P1	P4
Hàng đợi	P3	P3	P1 P2	P1 P2	P2 P4	P2 P4	P4 P1	P4 P1	P1 P2	P1 P2	P2 P4	P2 P4	P4 P1	P4 P1	P1	P1	P4	P4	

Tiến trình	P1	P3	P1	P2	P4	P1	P2	P4	P1	P4
Thời gian chạy	2	2	2	2	2	2	2	2	2	1
Thời gian chờ	0	2	4	6	8	10	12	14	16	

Thời gian chờ trung bình:

$$(10+4+2+10)/4 = 6.5$$

Bài 6: Xét trạng thái cấp phát tài nguyên của hệ thống như sau:

	X	Y	Z
P1	0	1	0
P2	2	0	0
P3	3	0	2
P4	2	1	1

Đã cấp

X	Y	Z
3	3	4

Còn lại

	X	Y	Z
P1	7	4	3
P2	1	2	2
P3	6	0	0
P4	0	1	1

Còn cần cấp

P2 yêu cầu cấp phát 1 tài nguyên Y, 2 tài nguyên Z. Sử dụng thuật toán người cho vay để xác định xem yêu cầu của P2 có được đáp ứng hay không?

P2 yêu cầu cấp phát:

	X	Y	Z
P2	0	1	2

Giả sử đã cấp:

	X	Y	Z
P1	0	1	0
P2	2	1	2
P3	3	0	2
P4	2	1	1

Còn Cần cấp:

	X	Y	Z
P1	7	4	3
P2	1	2	2
P3	6	0	0
P4	0	1	1

Ta xác định được W:

W	3	2	2
---	---	---	---

Lập bảng trạng thái:

	X	Y	Z
W	3	2	2
P2	5	3	4
P4	7	4	5
P1	7	5	5
P3	10	5	7

Kết luận: Trạng thái an toàn. Yêu cầu (0,1,2) đc chấp nhận.