

Ôn tập HĐH ngày  
29/11/2022

Hello Sussy Baka

# Đổi trang

## Kỹ thuật đổi trang

- ▶ Nếu không còn khung nào trống, HĐH chọn 1 khung đã cấp phát nhưng hiện không dùng tới và giải phóng khung này.
- ▶ Nội dung của khung sẽ được trao đổi ra đĩa, trang nhớ chứa trong khung sẽ được đánh dấu không được nằm trong bộ nhớ (bằng cách thay đổi bit P ở bảng trang).
- ▶ Khung được giải phóng cấp phát cho trang mới cần nạp vào

# Đổi trang

## Quá trình đổi trang:

- ▶ *Bước 1:* Xác định trang trên đĩa cần nạp vào MEM
- ▶ *Bước 2:* Nếu có khung trống trên MEM thì chuyển sang B4
- ▶ *Bước 3:*
  - Lựa chọn 1 khung trên MEM để giải phóng, theo 1 thuật toán nào đó
  - Ghi nội dung khung bị đổi ra đĩa (nếu cần), cập nhật bảng trang và bảng khung
- ▶ *Bước 4:* Đọc trang cần nạp vào khung vừa giải phóng; cập nhật bảng trang và bảng khung
- ▶ *Bước 5:* Thực hiện tiếp tiến trình từ điểm bị dừng trước khi đổi trang

# Đổi trang

## Đổi trang có ghi và đổi trang không ghi:

- ▶ Nếu nhu cầu đổi trang xuất hiện khi nạp trang mới, thời gian nạp trang sẽ tăng đáng kể do thêm nhu cầu ghi trạng bị đổi ra đĩa.
- ▶ Giúp nhận biết các trang không thay đổi từ lúc nạp và không ghi ngược ra đĩa
- ▶ Sử dụng thêm bit sửa đổi M trong khoản mục trang để đánh dấu trang đã bị sửa đổi (1) hay chưa (0)
- ▶ Các khung bị khóa:
  - Khi tìm các trang để giải phóng và đổi trang, HĐH sẽ trừ ra một số khung
  - Một số khung sẽ không bị giải phóng trong quá trình tìm kiếm khung để đổi trang là *các khung bị khóa*
  - VD: *Khung chứa tiến trình nhân của HĐH, chứa các cấu trúc thông tin điều khiển quan trọng*
  - Nhận biết bởi 1 bit riêng chứa trong khoản mục

# Đổi trang tối ưu (OPT):

- ▶ *Chọn trang sẽ không được dùng tới trong khoảng thời gian lâu nhất trong tương lai để đổi hay trang có lần sử dụng tiếp theo xa nhất*
- ▶ Cho phép giảm tối thiểu sự thiếu trang và tối ưu theo tiêu chuẩn này
- ▶ HĐH đoán trước được nhu cầu sử dụng các trang trong tương lai
- ▶ Không áp dụng trong thực tế chỉ so sánh với các chiến lược khác

# Đổi trang tối ưu (OPT):

*Chọn trang sẽ không được dùng tới trong khoảng thời gian lâu nhất trong tương lai để đổi hay trang có lần sử dụng tiếp theo xa nhất*

Giả sử tiến trình được cấp 3 khung, không gian nhớ của tiến trình có 5 trang và các trang của tiến trình được truy cập theo thứ tự như sau:

2, 3, 2, 1, 5, 2, 4, 5, 3, 2, 5, 2.

Xác định thứ tự nạp đổi trang sử dụng thuật toán tối ưu OPT.

	2	3	2	1	5	2	4	5	3	2	5	2																																				
OPT	<table><tr><td>2</td></tr><tr><td></td></tr><tr><td></td></tr></table>	2			<table><tr><td>2</td></tr><tr><td>3</td></tr><tr><td></td></tr></table>	2	3		<table><tr><td>2</td></tr><tr><td>3</td></tr><tr><td></td></tr></table>	2	3		<table><tr><td>2</td></tr><tr><td>3</td></tr><tr><td>1</td></tr></table>	2	3	1	<table><tr><td>2</td></tr><tr><td>3</td></tr><tr><td>5</td></tr></table> F	2	3	5	<table><tr><td>2</td></tr><tr><td>3</td></tr><tr><td>5</td></tr></table>	2	3	5	<table><tr><td>4</td></tr><tr><td>3</td></tr><tr><td>5</td></tr></table> F	4	3	5	<table><tr><td>4</td></tr><tr><td>3</td></tr><tr><td>5</td></tr></table>	4	3	5	<table><tr><td>4</td></tr><tr><td>3</td></tr><tr><td>5</td></tr></table>	4	3	5	<table><tr><td>2</td></tr><tr><td>3</td></tr><tr><td>5</td></tr></table> F	2	3	5	<table><tr><td>2</td></tr><tr><td>3</td></tr><tr><td>5</td></tr></table>	2	3	5	<table><tr><td>2</td></tr><tr><td>3</td></tr><tr><td>5</td></tr></table>	2	3	5
2																																																
2																																																
3																																																
2																																																
3																																																
2																																																
3																																																
1																																																
2																																																
3																																																
5																																																
2																																																
3																																																
5																																																
4																																																
3																																																
5																																																
4																																																
3																																																
5																																																
4																																																
3																																																
5																																																
2																																																
3																																																
5																																																
2																																																
3																																																
5																																																
2																																																
3																																																
5																																																

# Đổi trang tối ưu (OPT):

Giả sử tiến trình được cấp 3 khung, không gian nhớ của tiến trình có 5 trang và các trang của tiến trình được truy cập theo thứ tự như sau:

2, 3, 2, 1, 5, 2, 4, 5, 3, 2, 5, 2.

Xác định thứ tự nạp đổi trang sử dụng thuật toán tối ưu OPT.

*Chọn trang sẽ không được dùng tới trong khoảng thời gian lâu nhất trong tương lai để đổi hay trang có lần sử dụng tiếp theo xa nhất*

Trang đã có sẵn trong khung      Trang mới được đổi vào

2	3	2	1	5	2	4	5	3	2	5	2
2	2	2	2	2	2	4	4	4	2	2	2
	3	3	3	3	3	3	3	3	3	3	3
			1	5	5	5	5	5	5	5	5
				F		F			F		



# Đổi trang tối ưu (OPT):

**Câu 4:** Bộ nhớ vật lý có 3 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

*Chọn trang sẽ không được dùng tới trong khoảng thời gian lâu nhất trong tương lai để đổi hay trang có lần sử dụng tiếp theo xa nhất*





# Đổi trang tối ưu (OPT):

**Câu 4:** Bộ nhớ vật lý có 3 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

1	2	3	4	2	1	5	6	2	1	2
1	1	1	1	1	1	1	1	1	1	1
	2	2	2	2	2	2	2	2	2	2
		3	4	4	4	5	6	6	6	6
			F			F	F			

# Đổi trang tối ưu (OPT):

**Câu 4:** Bộ nhớ vật lý có 3 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

2	3	7	6	3	2	1	2	3	6
1	3	3	3	3	3	3	3	3	3
2	2	7	7	7	2	2	2	2	2
6	6	6	6	6	6	1	1	1	6
	F	F			F	F			F

# Vào trước ra trước (FIFO):

- ▶ Trang nào được nạp vào trước thì bị đổi ra trước
- ▶ Phương pháp đơn giản nhất
- ▶ Trang bị trao đổi là trang nằm lâu nhất trong bộ nhớ

# Vào trước ra trước (FIFO):

**Ví dụ:** Giả sử tiến trình được cấp 3 khung, không gian nhớ của tiến trình có 5 trang và các trang của tiến trình được truy cập theo thứ tự như sau: 2, 3, 2, 1, 5, 2, 4, 5, 3, 2, 5, 2

2	3	2	1	5	2	4	5	3	2	5	2
2	2	2	2	5	5	5	5	3	3	3	3
	3	3	3	3	2	2	2	2	2	5	5
			1	1	1	4	4	4	4	4	2
				F	F	F		F		F	F

# Vào trước ra trước (FIFO):

**Câu 4:** Bộ nhớ vật lý có 3 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

- ▶ Trang nào được nạp vào trước thì bị đổi ra trước



# Vào trước ra trước (FIFO):

**Câu 4:** Bộ nhớ vật lý có 3 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

1	2	3	4	2	1	5	6	2	1	2
1	1	1	4	4	4	4	6	6	6	6
	2	2	2	2	1	1	1	2	2	2
		3	3	3	3	5	5	5	1	1
			F			F	F	F	F	

# Vào trước ra trước (FIFO):

**Câu 4:** Bộ nhớ vật lý có 3 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

2	3	7	6	3	2	1	2	3	6
6	3	3	3	3	2	2	2	2	6
2	2	7	7	7	7	1	1	1	1
1	1	1	6	6	6	6	6	3	3
	F	F	F		F	F		F	F



# Vào trước ra trước (FIFO):

21

Giả sử tiến trình cấp 3 khung nhớ vật lý, các trang của tiến trình được truy cập theo thứ tự sau : 1,2,3,4,5,3,4,1,6,7,8,7,6,8. Xác định số lần đổi trang khi sử dụng chiến lược đổi trang FIFO: \*

(4 Điểm)



10:00

# Vào trước ra trước (FIFO):

21

Giả sử tiến trình cấp 3 khung nhớ vật lý, các trang của tiến trình được truy cập theo thứ tự sau : 1,2,3,4,5,3,4,1,6,7,8,7,6,8. Xác định số lần đổi trang khi sử dụng chiến lược đổi trang FIFO: \*

(4 Điểm)

1	2	3	4	5	3	4	1	6	7	8	7	6	8
1	1	1	4	4	4	4	4	6	6	6	6	6	6
	2	2	2	5	5	5	5	5	7	7	7	7	7
		3	3	3	3	3	1	1	1	8	8	8	8
			F	F			F	F	F	F			

# Đổi trang có lần sử dụng cuối cách lâu nhất (LRU)

- ▶ Trang bị đổi là trang mà thời gian từ lần truy cập cuối cùng đến thời điểm hiện tại là lâu nhất (lâu rồi không truy cập thì thay thế)
- ▶ Theo nguyên tắc cục bộ về thời gian, đó chính là trang ít có khả năng sử dụng tới nhất trong tương lai
- ▶ Thực tế LRU cho kết quả tốt gần như phương pháp đổi trang tối ưu

# Đổi trang có lần sử dụng cuối cách lâu nhất (LRU)

Sử dụng biến đếm:

- Mỗi khoản mục của bảng phân trang sẽ có thêm một trường chứa thời gian truy cập trang lần cuối - Là thời gian logic
- CPU cũng được thêm một bộ đếm thời gian logic này
- Chỉ số của bộ đếm tăng mỗi khi xảy ra truy cập bộ nhớ
- Mỗi khi một trang nhớ được truy cập, chỉ số của bộ đếm sẽ được ghi vào trường thời gian truy cập trong khoản mục của trang đó
- Trường thời gian luôn chứa thời gian truy cập trang lần cuối
- Trang bị đổi là trang có giá trị thời gian nhỏ nhất

# Đổi trang có lần sử dụng cuối cách lâu nhất (LRU)

## ► Sử dụng ngăn xếp

- Ngăn xếp đặc biệt được sử dụng để chứa các số trang
- Mỗi khi một trang nhớ được truy cập, số trang sẽ được chuyển lên đỉnh ngăn xếp
- Đỉnh ngăn xếp sẽ chứa trang được truy cập gần đây nhất
- Đáy ngăn xếp chính là trang LRU, tức là trang cần trao đổi
- Tránh tìm kiếm trong bảng phân trang
- Thích hợp thực hiện bằng phần mềm

# Đổi trang có lần sử dụng cuối cách lâu nhất (LRU)

- ▶ Trang bị đổi là trang mà thời gian từ lần truy cập cuối cùng đến thời điểm hiện tại là lâu nhất (lâu rồi không truy cập thì thay thế)

2	3	2	1	5	2	4	5	3	2	5	2																																				
<table><tr><td>2</td></tr><tr><td></td></tr><tr><td></td></tr></table>	2			<table><tr><td>2</td></tr><tr><td>3</td></tr><tr><td></td></tr></table>	2	3		<table><tr><td>2</td></tr><tr><td>3</td></tr><tr><td></td></tr></table>	2	3		<table><tr><td>2</td></tr><tr><td>3</td></tr><tr><td>1</td></tr></table>	2	3	1	<table><tr><td>2</td></tr><tr><td>5</td></tr><tr><td>1</td></tr></table> F	2	5	1	<table><tr><td>2</td></tr><tr><td>5</td></tr><tr><td>1</td></tr></table>	2	5	1	<table><tr><td>2</td></tr><tr><td>5</td></tr><tr><td>4</td></tr></table> F	2	5	4	<table><tr><td>2</td></tr><tr><td>5</td></tr><tr><td>4</td></tr></table>	2	5	4	<table><tr><td>3</td></tr><tr><td>5</td></tr><tr><td>4</td></tr></table> F	3	5	4	<table><tr><td>3</td></tr><tr><td>5</td></tr><tr><td>2</td></tr></table> F	3	5	2	<table><tr><td>3</td></tr><tr><td>5</td></tr><tr><td>2</td></tr></table>	3	5	2	<table><tr><td>3</td></tr><tr><td>5</td></tr><tr><td>2</td></tr></table>	3	5	2
2																																															
2																																															
3																																															
2																																															
3																																															
2																																															
3																																															
1																																															
2																																															
5																																															
1																																															
2																																															
5																																															
1																																															
2																																															
5																																															
4																																															
2																																															
5																																															
4																																															
3																																															
5																																															
4																																															
3																																															
5																																															
2																																															
3																																															
5																																															
2																																															
3																																															
5																																															
2																																															



# Đổi trang có lần sử dụng cuối cách lâu nhất (LRU)

**Câu 4:** Bộ nhớ vật lý có 3 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

Trang bị đổi là trang mà thời gian từ lần truy cập cuối cùng đến thời điểm hiện tại là lâu nhất (lâu rồi không truy cập thì thay thế)





# Đổi trang có lần sử dụng cuối cách lâu nhất (LRU)

**Câu 4:** Bộ nhớ vật lý có 3 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

1	2	3	4	2	1	5	6	2	1	2																																									
<table><tr><td>1</td></tr><tr><td></td></tr><tr><td></td></tr></table>	1			<table><tr><td>1</td></tr><tr><td>2</td></tr><tr><td></td></tr></table>	1	2		<table><tr><td>1</td></tr><tr><td>2</td></tr><tr><td>3</td></tr></table>	1	2	3	<table><tr><td>4</td></tr><tr><td>2</td></tr><tr><td>3</td></tr><tr><td>F</td></tr></table>	4	2	3	F	<table><tr><td>4</td></tr><tr><td>2</td></tr><tr><td>3</td></tr><tr><td></td></tr></table>	4	2	3		<table><tr><td>4</td></tr><tr><td>2</td></tr><tr><td>1</td></tr><tr><td>F</td></tr></table>	4	2	1	F	<table><tr><td>5</td></tr><tr><td>2</td></tr><tr><td>1</td></tr><tr><td>F</td></tr></table>	5	2	1	F	<table><tr><td>5</td></tr><tr><td>6</td></tr><tr><td>1</td></tr><tr><td>F</td></tr></table>	5	6	1	F	<table><tr><td>5</td></tr><tr><td>6</td></tr><tr><td>2</td></tr><tr><td>F</td></tr></table>	5	6	2	F	<table><tr><td>1</td></tr><tr><td>6</td></tr><tr><td>2</td></tr><tr><td>F</td></tr></table>	1	6	2	F	<table><tr><td>1</td></tr><tr><td>6</td></tr><tr><td>2</td></tr><tr><td></td></tr></table>	1	6	2	
1																																																			
1																																																			
2																																																			
1																																																			
2																																																			
3																																																			
4																																																			
2																																																			
3																																																			
F																																																			
4																																																			
2																																																			
3																																																			
4																																																			
2																																																			
1																																																			
F																																																			
5																																																			
2																																																			
1																																																			
F																																																			
5																																																			
6																																																			
1																																																			
F																																																			
5																																																			
6																																																			
2																																																			
F																																																			
1																																																			
6																																																			
2																																																			
F																																																			
1																																																			
6																																																			
2																																																			
<table><tr><td>1</td></tr><tr><td></td></tr><tr><td></td></tr></table>	1			<table><tr><td>2</td></tr><tr><td>1</td></tr><tr><td></td></tr></table>	2	1		<table><tr><td>3</td></tr><tr><td>2</td></tr><tr><td>1</td></tr></table>	3	2	1	<table><tr><td>4</td></tr><tr><td>3</td></tr><tr><td>2</td></tr></table>	4	3	2	<table><tr><td>2</td></tr><tr><td>4</td></tr><tr><td>3</td></tr></table>	2	4	3	<table><tr><td>1</td></tr><tr><td>2</td></tr><tr><td>4</td></tr></table>	1	2	4	<table><tr><td>5</td></tr><tr><td>1</td></tr><tr><td>2</td></tr></table>	5	1	2	<table><tr><td>6</td></tr><tr><td>5</td></tr><tr><td>1</td></tr></table>	6	5	1	<table><tr><td>2</td></tr><tr><td>6</td></tr><tr><td>5</td></tr></table>	2	6	5	<table><tr><td>1</td></tr><tr><td>2</td></tr><tr><td>6</td></tr></table>	1	2	6	<table><tr><td>2</td></tr><tr><td>1</td></tr><tr><td>6</td></tr></table>	2	1	6								
1																																																			
2																																																			
1																																																			
3																																																			
2																																																			
1																																																			
4																																																			
3																																																			
2																																																			
2																																																			
4																																																			
3																																																			
1																																																			
2																																																			
4																																																			
5																																																			
1																																																			
2																																																			
6																																																			
5																																																			
1																																																			
2																																																			
6																																																			
5																																																			
1																																																			
2																																																			
6																																																			
2																																																			
1																																																			
6																																																			

# Đổi trang có lần sử dụng cuối cách lâu nhất (LRU)

**Câu 4:** Bộ nhớ vật lý có 3 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

2	3	7	6	3	2	1	2	3	6																																							
<table><tr><td>1</td></tr><tr><td>6</td></tr><tr><td>2</td></tr></table>	1	6	2	<table><tr><td>1</td></tr><tr><td>3</td></tr><tr><td>2</td></tr><tr><td>F</td></tr></table>	1	3	2	F	<table><tr><td>7</td></tr><tr><td>3</td></tr><tr><td>2</td></tr><tr><td>F</td></tr></table>	7	3	2	F	<table><tr><td>7</td></tr><tr><td>3</td></tr><tr><td>6</td></tr><tr><td>F</td></tr></table>	7	3	6	F	<table><tr><td>7</td></tr><tr><td>3</td></tr><tr><td>6</td></tr><tr><td>F</td></tr></table>	7	3	6	F	<table><tr><td>2</td></tr><tr><td>3</td></tr><tr><td>6</td></tr><tr><td>F</td></tr></table>	2	3	6	F	<table><tr><td>2</td></tr><tr><td>3</td></tr><tr><td>1</td></tr><tr><td>F</td></tr></table>	2	3	1	F	<table><tr><td>2</td></tr><tr><td>3</td></tr><tr><td>1</td></tr><tr><td></td></tr></table>	2	3	1		<table><tr><td>2</td></tr><tr><td>3</td></tr><tr><td>1</td></tr><tr><td></td></tr></table>	2	3	1		<table><tr><td>2</td></tr><tr><td>3</td></tr><tr><td>6</td></tr><tr><td>F</td></tr></table>	2	3	6	F
1																																																
6																																																
2																																																
1																																																
3																																																
2																																																
F																																																
7																																																
3																																																
2																																																
F																																																
7																																																
3																																																
6																																																
F																																																
7																																																
3																																																
6																																																
F																																																
2																																																
3																																																
6																																																
F																																																
2																																																
3																																																
1																																																
F																																																
2																																																
3																																																
1																																																
2																																																
3																																																
1																																																
2																																																
3																																																
6																																																
F																																																
<table><tr><td>2</td></tr><tr><td>1</td></tr><tr><td>6</td></tr></table>	2	1	6	<table><tr><td>3</td></tr><tr><td>2</td></tr><tr><td>1</td></tr></table>	3	2	1	<table><tr><td>7</td></tr><tr><td>3</td></tr><tr><td>2</td></tr></table>	7	3	2	<table><tr><td>6</td></tr><tr><td>7</td></tr><tr><td>3</td></tr></table>	6	7	3	<table><tr><td>3</td></tr><tr><td>6</td></tr><tr><td>7</td></tr></table>	3	6	7	<table><tr><td>2</td></tr><tr><td>3</td></tr><tr><td>6</td></tr></table>	2	3	6	<table><tr><td>1</td></tr><tr><td>2</td></tr><tr><td>3</td></tr></table>	1	2	3	<table><tr><td>2</td></tr><tr><td>1</td></tr><tr><td>3</td></tr></table>	2	1	3	<table><tr><td>3</td></tr><tr><td>2</td></tr><tr><td>1</td></tr></table>	3	2	1	<table><tr><td>6</td></tr><tr><td>3</td></tr><tr><td>2</td></tr></table>	6	3	2									
2																																																
1																																																
6																																																
3																																																
2																																																
1																																																
7																																																
3																																																
2																																																
6																																																
7																																																
3																																																
3																																																
6																																																
7																																																
2																																																
3																																																
6																																																
1																																																
2																																																
3																																																
2																																																
1																																																
3																																																
3																																																
2																																																
1																																																
6																																																
3																																																
2																																																

# CLOCK: Thuật toán đồng hồ

Mỗi trang được gắn thêm 1 bit sử dụng U

- Khi trang được truy cập đọc/ ghi:  $U = 1$
- Ngay khi trang được đọc vào bộ nhớ:  $U = 1$

Các khung/ trang có thể bị đổi được liên kết vào 1 danh sách vòng

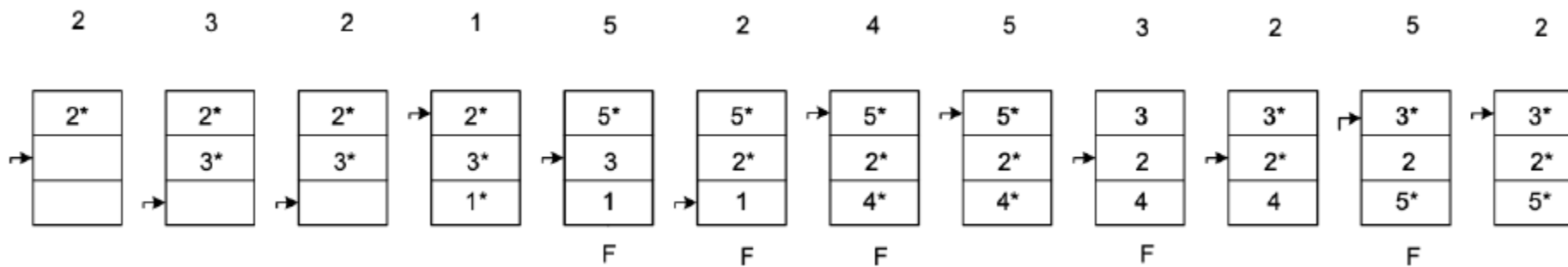
Khi một trang nào đó bị đổi, con trỏ được dịch chuyển để trỏ vào trang tiếp theo trong danh sách

Khi có nhu cầu đổi trang, HDH kiểm tra trang đang bị trỏ tới

- Nếu  $U=0$ : trang sẽ bị đổi ngay
- Nếu  $U=1$ : đặt  $U=0$  và trỏ sang trang tiếp theo, lặp lại quá trình

U của tất cả các trang trong danh sách  $=1$  con trỏ sẽ quay đúng 1 vòng, đặt U của các trang  $=0$  và trang hiện thời bị trỏ sẽ bị đổi

# CLOCK: Thuật toán đồng hồ



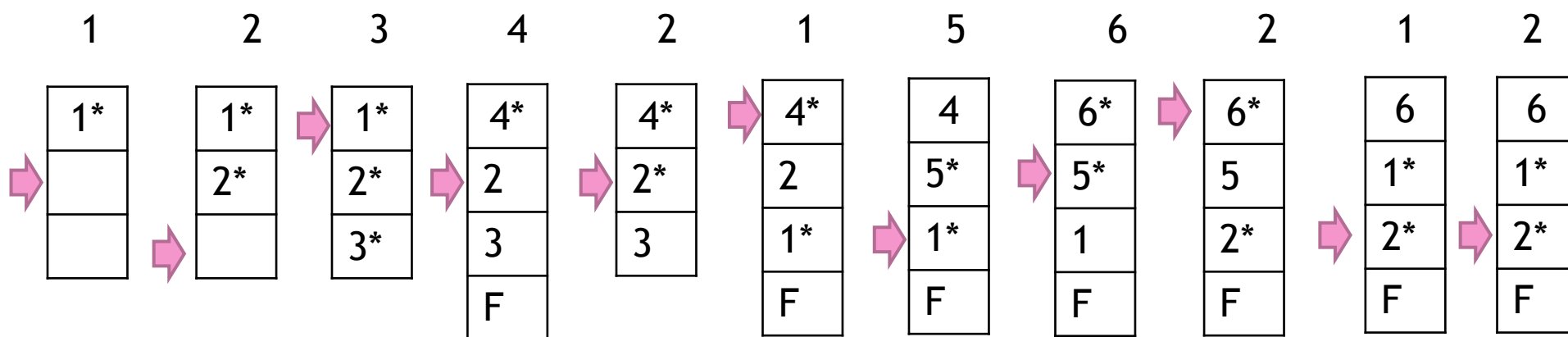
# CLOCK: Thuật toán đồng hồ

**Câu 4:** Bộ nhớ vật lý có 3 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



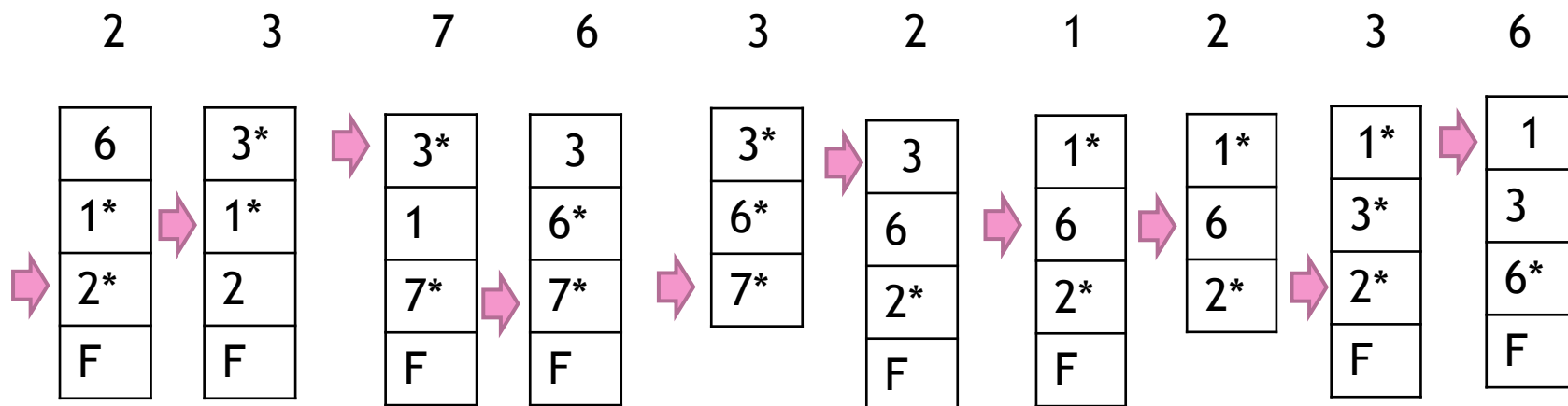
# CLOCK: Thuật toán đồng hồ

**Câu 4:** Bộ nhớ vật lý có 3 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



# CLOCK: Thuật toán đồng hồ

**Câu 4:** Bộ nhớ vật lý có 3 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





# BTVN

**Câu 5:** Giả sử tiến trình được cấp 4 khung nhớ vật lý, các trang của tiến trình được truy cập theo thứ tự sau :

1,2,3,4,5,3,4,1,6,7,8,7,8,9,7,8,9,5. Hãy xác định thứ tự nạp và đổi trang nếu sử dụng:

- Thuật toán tối ưu
- FIFO
- LRU
- Đồng hồ