
LRU 实现 & Belady

LRU 实现

1. 链表

- ① 最久的页：链表尾
- ② 最新访问页：链表首
- ③ 当目前要访问的页存在于链表时，移动该结点至链表首
- ④ 当页表满需要替换时删除链表尾

查找的开销，移动

LRU 实现

2. 堆栈

- ① 最久的页：栈低
- ② 最新访问页：栈顶
- ③ 当目前要访问的页存在于栈时，抽出并移至栈顶
- ④ 当页表满需要替换时删除栈低页帧

查找的开销，移动

LRU 实现

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

FIFO Belady

- 页帧增加，缺页减少

- 2 3 2 1 5 2 4 5 3 2 5 2

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

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

FIFO Belady

- 页帧增加，缺页增加

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

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

FIFO Belady

- Belady 现象

- 采用 FIFO 算法时，一个进程 P 要访问 M 个页，OS 分配 N ($M > N$) 个内存页面给进程 P ；对于一个访问序列 S ，发生缺页次数为 $PE(S, N)$ 。当 N 增大时， $PE(S, N)$ 时大时小。
- 产生原因：FIFO 算法的置换特征与进程访问内存的动态特征是矛盾的。因为较早调入的页可能是会被经常被访问的页，这些页在 FIFO 算法下被反复调入和调出，并且有 Belady 现象。
- 与抖动现象不同