1.
   - 可能发生：
     - TLB未命中，没有缺页错误：页码不在TLB中，但是可以在页表中查询到页码对应的有效页表项。
     - TLB未命中，发生缺页错误：页码不在TLB中，需要查询页表，但是页表中也没有页码的有效页表项
     - TLB命中，没有发生缺页错误：有效的页码和对应的页帧信息在TLB中，说明查询的页在内存中，没有缺页错误
   - 不可能发生：
     - TLB命中，发生缺页错误：TLB命中可以得到页码的对应的有效物理页帧信息，不会发生缺页错误

2. 3个帧的请求调页：7，2，3，1，2，5，3，4，6，7，7，1，0，5，4，6，2，3，0，1
   - LRU置换：
     看过去最近使用谁离得远

     18次缺页错误

7	**7**	**7**	**1**	**1**	**1**	**3**	**3**
	2	2	2	2	2	2	4
		3	3	3	5	5	5

3	**7**	**7**	**7**	**7**	**5**	**5**	**5**
4	4	4	1	1	1	4	4
6	6	6	6	0	0	0	6

2	**2**	**2**	**1**				
4	3	3	3				
6	6	0	0				

   - FIFO：
     看最新一次进入后停留时间最长

     17次缺页

7	**7**	**7**	**1**	**1**	**1**	**1**	**1**
	2	2	2	2	5	5	5
		3	3	3	3	3	4

6	**6**	**6**	**6**	**0**	**0**	**0**	**6**
5	7	7	7	7	5	5	5

6	**6**	**6**	**6**	**0**	**0**	**0**	**6**
4	4	4	1	1	1	4	4

6	**6**	**0**	**0**				
2	2	2	1				
4	3	3	3				

- 最优置换算法

  7，2，3，1，2，5，3，4，6，7，7，1，0，5，4，6，2，3，0，1

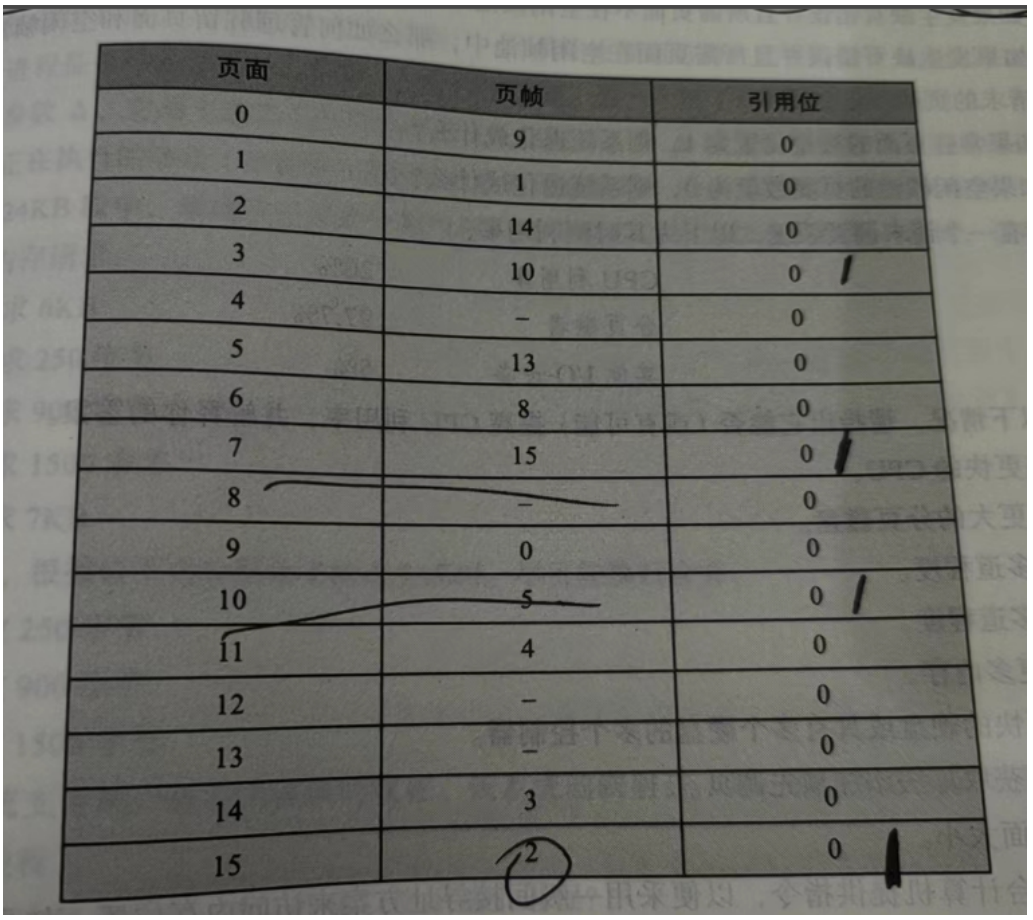
  看未来哪个最久才被二次使用
  缺页：13次

7	**7**	**7**	**1**	**1**	**1**	**1**	**1**
	2	2	2	2	5	5	5
		3	3	3	3	3	4

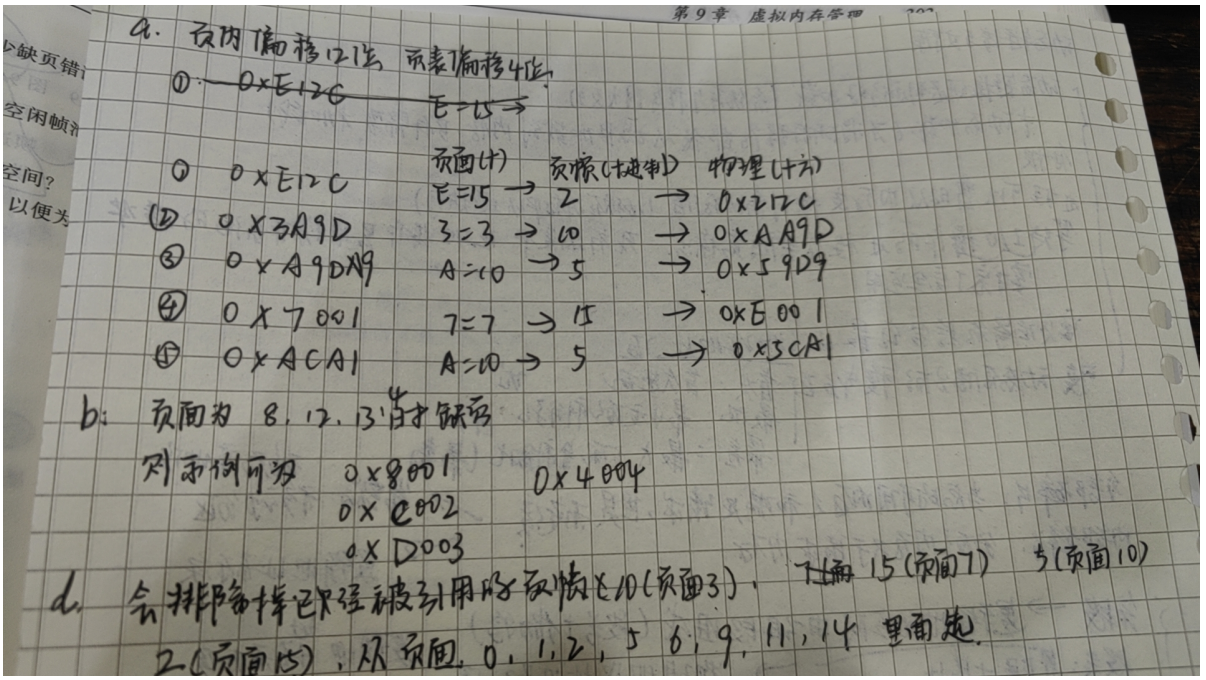
1	**1**	**1**	**1**	**1**	**1**	**1**	**1**
5	5	5	5	5	5	4	6
6	7	7	7	0	0	0	0

1	**1**	**1**	**1**				
2	3	3	3				
0	0	0	0				

3.



页面	页帧	引用位
0	9	0
1	1	0
2	14	0
3	10	0
4	–	0
5	13	0
6	8	0
7	15	0
8	–	0
9	0	0
10	5	0
11	4	0
12	–	0
13	–	0
14	3	0
15	2	0



4. ○ 系统抖动的原因：
   - 系统中已经存在的进程发生缺页错误，需要更多的帧，于是从其他进程那里置换帧或者从自己未被引用帧中选取帧置换。
   - 由于其他进程也需要这些页面，其他进程也会发生缺页错误。
   - 此时系统内的进程将大部分时间用于使用或等待调页设备将页面换出换入，CPU利用率下降。
   - 操作系统看到下降的CPU利用率负反馈调节，增加多道程序，新进程和旧进程一起竞争换入换出有限的页帧，导致进程调页时间多于执行时间，发生系统抖动

- 系统抖动的检测：
  - 计算进程每个工作集大小的总和也就是进程帧的需求总量，当它大于可用帧总数，将发生系统抖动
  - 检测进程缺页错误率，抖动具有高缺页错误率
- 消除问题：
  - 工作集：给进程分配大于工作集的可用帧。如果工作集大小总和增加，超过可用帧总数，将另外一个进程挂起，将其页面写出，帧重新分配给帧不够的进程
  - 缺页错误率：设置缺页错误率的上下限，缺页错误率高于上限，再分配一帧，低于下限，再删除一帧。缺页错误率增加但是没有可用空闲帧了，选择进程挂起，释放帧重新分配。

5.

- 顺序文件访问：
  连续分配：顺序文件访问上可以减少寻道时间，但是存在着外部碎片，

  链接分配：支持顺序访问，顺序搜索链表

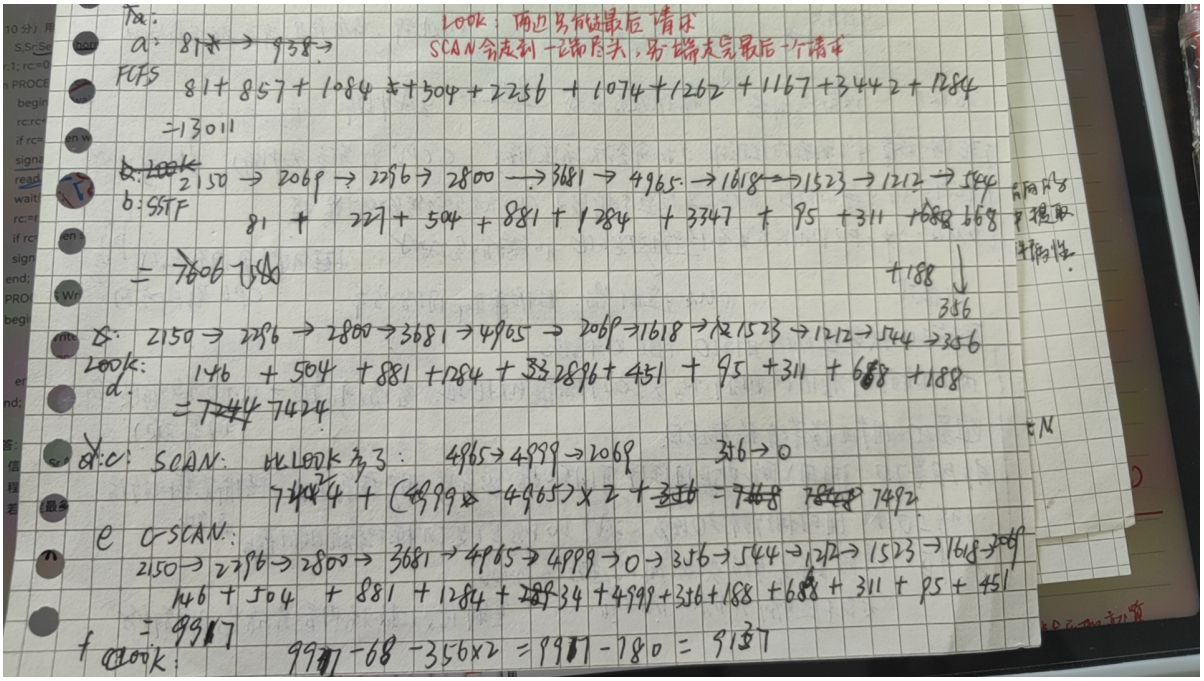
  索引分配：支持顺序访问，查询索引块上的目录，但是索引块的开销可能比链接分配的指针开销大，如果有多级索引性能会下降。
- 随机文件访问：

  连续分配：支持随机访问，可以在内存记录保存文件开始地址，由于连续分配，可以立即计算第i块的磁盘地址

  链接分配：对第i块的访问可能要读取i次磁盘，不有效支持随机访问

  索引分配：支持随机访问，查询索引块，还可以将索引块缓存在内存中，对第i块的直接访问可能只需要一次磁盘读取，取决于文件结构和文件大小和索引结构

6.



7. 减少随机访问文件的开销。链式索引中链接文件的块与块的指针在每个块中，随机访问时，需要从文件的第一个块顺序搜索，为了得到下一个块的指针必须要磁盘读当前块，如果要访问文件的第i个块，就要磁盘读i次。但是FAT表中每个块有一个条目，链接块与块之间的指针全部存储在Fat表中，每个条目包含了指向下一个磁盘块在Fat表的位置的指针，随机访问文件时，只需要查询FAT表顺序搜索定位到指定磁盘块。如果FAT表在磁盘中只需要两次磁盘读，在内存缓存中只需要一次磁盘读，大大降低开销。