

作业 9

9.1 假设一台计算机上运行的一个进程其地址空间有 8 个虚页（每个虚页大小为 4KB，页号为 1 至 8），操作系统给该进程分配了 4 个物理页框（每个页框大小为 4KB），该进程对地址空间中虚页的访问顺序为 1 2 4 5 6 3 7 3 7 8。假设分配给进程的 4 个物理页框初始为空，请计算：

（1）如果操作系统采用 CLOCK 算法管理内存，那么该进程访存时会发生多少次 page fault？当进程访问完上述虚页后，物理页框中保存的是哪些虚页？

（2）如果操作系统采用 LRU 算法管理内存，请再次回答（1）中的两个问题。请回答虚页保存情况时，写出 LRU 链的组成，标明 LRU 端和 MRU 端。

9.2 假设一台计算机给每个进程都分配 4 个物理页框，每个页框大小为 1KB。现有一个程序对一个二维整数数组（`uint32 X[64][64]`）进行赋值操作，该程序的代码段占用一个固定的页框，并一直存储在内存中。程序使用剩余 3 个物理页框存储数据。该程序操作的数组 X 以列存储形式保存在磁盘上，即 X[0][0]后保存的是 X[1][0]、X[2][0]...X[63][0]，然后再保存 X[0][1]，以此类推。当程序要赋值时，如果所赋值的数组元素不在内存中，则会触发 page fault，操作系统将相应元素以页框粒度交换至内存。如果该进程的物理页框已经用满，则会进行页换出。该程序有如下两种写法。

写法 1：

```
for(int i=0;i<64;i++)  
    for(int j=0;j<64;j++)  
        X[i][j] = 0
```

写法 2：

```
for(int j=0;j<64;j++)  
    for(int i=0;i<64;i++)  
        X[i][j] = 0
```

请分析使用这两种写法时，各自会产生多少次 page fault？（注：请写出分析或计算过程）

9.3 假设一个程序有两个段，其中段 0 保存代码指令，段 1 保存读写的数据。段 0 的权限是可读可执行，段 1 的权限是可读可写，如下所示。该程序运行的内存系统提供的虚址空间为 14-bit 空间，其中低 10-bit 为页内偏移，高 4-bit 为页号。

Segment 0		Segment 1	
Read/Execute		Read/Write	
Virtual Page #	Page frame #	Virtual Page #	Page frame #
0	2	0	On Disk
1	On Disk	1	14
2	11	2	9
3	5	3	6
4	On Disk	4	On Disk
5	On Disk	5	13
6	4	6	8
7	3	7	12

当有如下的访存操作时，请给出每个操作的实际访存物理地址或是产生的异常类型（例如缺页异常、权限异常等）

- (1) 读取段 1 中 page 1 的 offset 为 3 的地址
- (2) 向段 0 中 page 0 的 offset 为 16 的地址写入
- (3) 读取段 1 中 page 4 的 offset 为 28 的地址
- (4) 跳转至段 1 中 page 3 的 offset 为 32 的地址

9.4 假设一个程序对其地址空间中虚页的访问序列为 0,1,2, ..., 511, 422, 0, 1, 2, ..., 511, 333, 0, 1, 2, ..., 即访问一串连续地址（页 0 到页 511）后会随机访问一个页（页 422 或页 333），且这个访问模式会一直重复。请分析说明：

(1) 假设操作系统分配给该程序的物理页框为 500 个，那么，LRU, Second Chance 和 FIFO 这三种算法中哪一个会表现较好（即提供较高的命中率），或是这三种算法都表现不佳？为什么？

9.5 现有一个内存空间分配器，采用伙伴算法。假设物理内存总共 64 KB，

- 1) 请给出第一级的一对伙伴块的起始地址
- 2) 请给出第二级的二对伙伴块的起始地址
- 3) 地址 0xa700，已知它位于第 7 级伙伴块中，请问该块的伙伴块的起始地址