**作业10**

**唐嘉良**

**2020K8009907032**

**10.1 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**

答：64KB = 2^16B,假设内存空间按字节寻址。

1）第一级伙伴块起始地址分别为0x0,0x8000

2）第二级两对伙伴块起始地址分别为0x0,0x4000和0x8000,0xc000

3）第七级伙伴块每块的大小为64/2^7 = 512B，于是包含0xa700地址的最大七级伙伴块起始地址为0xa600

**10.2 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（即提供较高的缓存命中率），或是这三种算法都表现不佳？为什么？**

答：（1）LRU算法下，替换最长时间没有使用的页，所以每次循环（513次访问，前512次从0到511，最后一次随机访问）前500次缺页，然后12次也全部缺页，把0-11替换出来，最后一次随机访问的时候命中率有500/512。后面的循环中，如果之前命中，则前500个全部缺页，然后的12个也全部缺页......循环往复，缺页率非常高，即便之前不命中，也是一样的情况。该算法表现很差。

FIFO下，替换最老的页，第一次循环和LRU算法行为一致，区别在于随机数命中之后命中页的优先级变化了，但是之后的循环中依然会发生类似LRU算法频率的缺页，该算法表现也很差。

SecondChance下，区别在于后12次缺页时要消耗全部页框一次机会，随机数的时候也与FIFO类似，但是访问序列里没什么频繁访问的页，无法发挥该算法的优势，缺页率依旧很高。该算法表现依然很差，甚至消耗的资源还更多！

三种算法都表现不好。