**存储管理**

1、实验目的

存储管理的主要功能之一是合理地分配空间。请求页式管理是一种常用的虚拟存储管理技术。

本实验的目的是通过请求页式存储管理中页面置换算法模拟设计，了解虚拟存储技术的技术特点，掌握请求页式存储管理的页面置换算法。

2、实验内容

（1）通过随机数产生一个指令序列，共320条指令。指令的地址按下述原则生成：

①50%的指令是顺序执行的；

②25%的指令是均匀分布在前地址部分；

③25%的指令是均匀分布在后地址部分。

具体的实施方法是：

①在 [0，319] 的指令之间随即选取一起点m;

②顺序执行一条指令，即执行地址为m+1的指令；

③在前地址[0，m+1]中随机选取一条指令并执行，该指令的地址为m′；

④顺序执行一条指令，其地址为 m′+ 1；

⑤在后地址[m′+ 2，319]中随机选取一条指令并执行；

⑥重复上述步骤①-⑤，直到执行320次指令。

（2）将指令序列变换为页地址流

设：①页面大小为1k；

②用户内存容量为4页到32页；

③用户虚存容量为32k。

在用户虚存中，按每k存放10条指令排在虚存地址，即320条指令在虚存中的存放方式为：

第0条-第9条指令为第0页（对应虚存地址为[0，9]）；

第10条-第19条指令为第一页（对应虚存地址为[10，19]）；

… …

第310条~第319条指令为第31页（对应虚地址为[310，319]）。

按以上方式，用户指令可组成32页。

（3）计算并输出下述各种算法在不同内存容量下的命中率。

①先进先出的算法（FIFO）；

②最近最少使用算法（LRU)；

③最佳淘汰算法（OPT），先淘汰最不常用的页地址；

④最近未使用算法（NRU）。

命中率=1-页面失效次数/页地址流长度

在本实验中，页地址流长度为320，页面失效次数为每次访问相应指令时，该指令所对应的页不在内存的次数。

3、随机数产生办法，Linux或UNIX系统提供函数strand()和rand(),分别进行初始化和产生随机数。例如：

srand ();

语句可初始化一个随机数；

a[0]=10\*rand()/65535\*319+1;

a[1]=10\*rand()/65535\*a[0];

语句可用来产生a[0]与a[1]中的随机数。