《内存管理》实验指导书

【实验目的】

理解操作系统虚拟存储管理的原理，理解程序执行局部性的概念。

【实验内容】

设计一个虚拟存储区和内存工作区,并使用下列算法计算访问命中率。

（1） 进先出的算法（FIFO）

（2） 最近最少使用的算法（LRU）

（3） 最佳淘汰算法(OPT)

命中率=(1-页面失效次数)/页地址流长度

【实验背景】

【实验环境】

VC++6.0, CONSOLE 程序

【实验要求】

1、 理解 FIFO,LRU 算法原理，理解参考程序的原理和实现思路。

2、 完成程序的设计，重点完成 FIFO,LRU 算法

3、 分析运算结果，在分配不同的物理块情况下，各算法的缺页情况有什么规律？

4、 完成 OPT 算法

【程序设计指导】

本实验的程序设计基本上按照实验内容进行。即首先用 srand()和 rand()函数定义和产生指令序列，然后将指令序列变换成相应的页地址流，并针对不同的算法计算出相应的命中率。**实验中我们产生 320 条指令，每个虚拟页存放 10 条指令。进程分配的物理块从 4 变化到 32。**

【重要变量、结构和函数的建议】

(1)页面类型

```
typedef struct
{
    int pn;
    int pfn;
    int counter;
```

```
    int time
} pl-type;
```

其中 pn 为页号,pfn 为面号, counter 为一个周期内访问该页面的次数, time 为访问时间。

(2) 页面控制结构

```
pfc_struct
{
       int pn,pfn;
       struct pfc_struct *next;
}
typedef   struct pfc_struct pfc_type;
pfc_type  pfc_struct[total_vp], *freepf_head, *busypf_head;
pfc_type  *busypf_tail;
```

其中:

pfc[total_vp]定义用户进程虚页控制结构。

*freepf_head 为空页面头的指针。

*busypf_head 为忙页面头的指针。

*busypf_tail 为忙页面尾的指针。

(3) 函数定义

void initialize( ):初始化函数,给每个相关的页面赋值。

void FIFO( ):计算使用 FIFO 算法时的命中率。

void LRU( ):计算使用 LRU 算法时的命中率。

(4) 变量定义

int a[total_instruction]: 指令流数据组。

int page[total_instruction]: 每条指令所属的页号。

int offset[total_instruction]: 每页装入 10 条指令后取模运算页号偏移值。

int total_pf: 用户进程的内存页面数。

int disaffect: 页面失效次数。

【参考资料】

1、 MSDN

2、 教材和课件