《计算机操作系统》实验报告

实验题目：请求页式存储管理

姓名：严昕宇　　学号：20121802　　实验日期：2022.12.16

实验环境：

实验设备：Lenovo Thinkbook16+ 2022

开发环境：CLion 2022.3

实验目的：

近年来，由于大规模集成电路(LSI)和超大规模集成电路(VLSI)技术的发展，使存储器的容量不断扩大，价格大幅度下降。但从使用角度看，存储器的容量和成本总受到一定的限制。所以，提高存储器的效率始终是操作系统研究的重要课题之一。虚拟存储技术是用来扩大内存容量的一种重要方法。学生应独立地用高级语言编写几个常用的存储分配算法，并设计一个存储管理的模拟程序，对各种算法进行分析比较，评测其性能优劣，从而加深对这些算法的了解。

实验要求：

为了比较真实地模拟存储管理，可预先生成一个大致符合实际情况的指令地址流。然后模拟这样一种指令序列的执行来计算和分析各种算法的访问命中率。

实验内容：

1. 题目

本示例是采用页式分配存储管理方案，并通过分析计算不同页面淘汰算法情况下的访问命中率来比较各种算法的优劣。另外也考虑到改变页面大小和实际存储器容量对计算结果的影响，从而可为算则好的算法、合适的页面尺寸和实存容量提供依据。

本程序是按下述原则生成指令序列的：

(1) 50%的指令是顺序执行的。

(2) 25%的指令均匀散布在前地址部分。

(3) 25%的指令均匀散布在后地址部分。

示例中选用最佳淘汰算法(OPT)和最近最少使用页面淘汰算法(LRU)计算页面命中率。

公式为：

$$命中率 = 1 - \frac{页面失效次数}{页地址流长度}$$

假定虚存容量为 32K，页面尺寸从 1K 至 8K，实存容量从 4 页至 32 页。

2. 算法与框图

(1) 最佳淘汰算法(OPT)。

这是一种理想的算法，可用来作为衡量其他算法优劣的依据，在实际系统中是难以实现的，因为它必须先知道指令的全部地址流。由于本示例中已预先生成了全部的指令地址流，故可计算出最佳命中率。

该算法的准则是淘汰已满页表中不再访问或是最迟访问的的页。这就要求将页表中的页

逐个与后继指令访问的所有页比较，如后继指令不在访问该页，则把此页淘汰，不然得找出后继指令中最迟访问的页面淘汰。可见最佳淘汰算法要花费较长的运算时间。

(2) 最近最少使用页淘汰算法(LRU)。

这是一种经常使用的方法，有各种不同的实施方案，这里采用的是不断调整页表链的方法，即总是淘汰页表链链首的页，而把新访问的页插入链尾。如果当前调用页已在页表内，则把它再次调整到链尾。这样就能保证最近使用的页，总是处于靠近链尾部分，而不常使用的页就移到链首，逐个被淘汰，在页表较大时，调整页表链的代价也是不小的。

(3) 程序框图如下图 5 示。

图 5 计算页面命中率框图

3. 程序运行结果格式

(1) 程序运行结果格式

```
THE VIRTUAL ADDRESS STREAM AS FOLLOWS:
a[0] =16895          a[1]=16896          a[2]=16897          a[3]=16302
a[4]=25403          a[5]=13941          a[6]=13942          a[7]=8767
……
……
……
A[252]=23583          a[253]=20265          a[254]=20266          a[255]=20267
==========================================
The algorithm is:opt
PAGE NUMBER WITH SIZE 1k FOR EACH ADDRESS IS:
pageno[0]=17        pageno[1]=17        pageno[2]=17        pageno[3]=16
pageno[4]=25        pageno[5]=14        pageno[6]=14        pageno[7]=9
……
……
……
pageno[252]=24        pageno[253]=20        pageno[254]=20        pageno[255]=20

vmsize=32k        pagesize=1k
---------- ----------
```

```
------------ --------------
page assigned                    pages_in/total references
4                                7.0312500000000000E-1
6                                7.5781250000000000E-1
8                                8.0859375000000000E-1
10                               8.5546875000000000E-1
12                               8.9453125000000000E-1
14                               9.1406250000000000E-1
16                               9.1406250000000000E-1
18                               9.1406250000000000E-1
20                               9.1406250000000000E-1
22                               9.1406250000000000E-1
24                               9.1406250000000000E-1
26                               9.1406250000000000E-1
28                               9.1406250000000000E-1
30                               9.1406250000000000E-1
32                               9.1406250000000000E-1
PAGE NUMBER WITH SIZE 2k EACH ADDRESS IS:
……
……
……
PAGE NUMBER WITH SIZE 4k EACH ADDRESS IS:
……
……
……
PAGE NUMBER WITH SIZE 8k EACH ADDRESS IS:
……
……
……
End the result for opt
** ********************************************
the algorithm is lru
……
……
……同上
End the result for lru
*********************************************
```

(2) 说示例中使用的有关数据结构、常量和变量说明如下：

- length：被调试的指令地址流长度，可作为一个常量设定。
- called：当前请求的页面号。
- pagefault：页面失效标志，如当前请求页 called 已在页表内，则置 pagefault=false，否则为 true。
- table：页表。table[i]=j,表示虚存的第 j 页在实存的第 i 页中。
- used：当前被占用的实存页面数，可用来判断当前实存中是否有空闲页。

(3) 本程序启动后，屏幕上显示“the algorithm is：”，用户可选择最佳淘汰算法（打入“OPT”）或者最近最少使用淘汰算法(打入“LRU”) 计算页面命中率。当然还可以加入各种其他的算法。

4. 小结

(1) 编制评测各种算法性能的模拟程序是研制系统程序，尤其是操作系统所必须的。模拟的环境愈是真实，其结果愈是可靠，也就更有利于选择合适的方案。本实习虽属简单，但可作为一个尝试。

(2) 注意正整数的范围只能从 0……32767，限制程序中的虚存尺寸为 32K，实际如采用更大的虚存实存，更能说明问题。

结果：

- 基本分页式地址变换

```
------实验3-1：基本分页式地址变换------
页表如下：
===========
页号      块号
0         2
1         3
2         6
3         8
4         9
5         12
===========
---------------------------------


---------------------------------
页面结构如下：
15          11                  0

---------------------------------
|页号（4位）|位移量（12位）       |
---------------------------------

请输入逻辑地址(十六进制数)，并以'#'结束：
1234#
逻辑地址的二进制数为：
0001 0010 0011 0100
转换的物理地址的二进制数为：
0011 0010 0011 0100
物理地址的十六进制数为：
3234
```

- 页面置换算法

此处以第四、五章课程作业中的例题作为数据来源，对作业以及程序的正确性进行检查。

【题目】在一个请求分页存储管理系统中，一个作业的页面走向为 4，3，2，1，4，3，5，4，3，2，1，5，当分配给作业的物理块数分别为 3 和 4 时，试计算采用下述页面淘汰算法时的缺页率 (假设，开始执行时主存中没有页面)，并比较结果。(给出置换和计算过程)

(1) 最佳置换算法 (OPT)

理论计算如下：

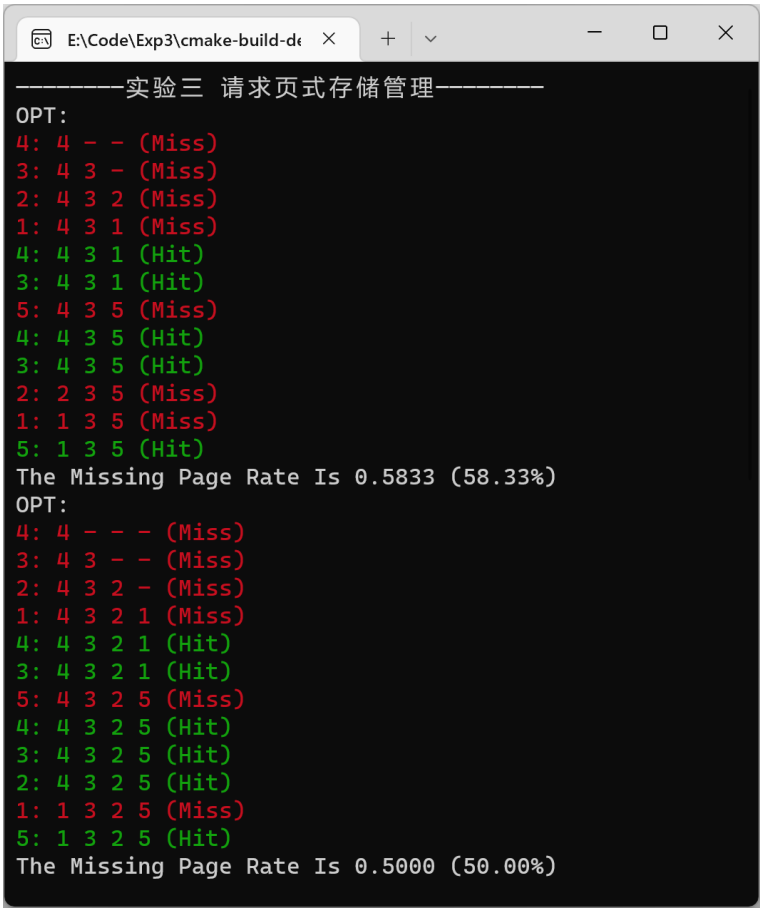
物理块数 m=3

访问页面	4	3	2	1	4	3	5	4	3	2	1	5
内存块 1	4	4	4	4			4			2	1	
内存块 2		3	3	3			3			3	3	
内存块 3			2	1			5			5	5	
是否缺页	√	√	√	√			√			√	√	

物理块数 m=4

访问页面	4	3	2	1	4	3	5	4	3	2	1	5
内存块 1	4	4	4	4			4				1	
内存块 2		3	3	3			3				3	
内存块 3			2	2			2				2	
内存块 4				1			5				5	
是否缺页	√	√	√	√			√				√	

程序输出如下：



(2) 最近最久未使用算法 (LRU)

理论计算如下：

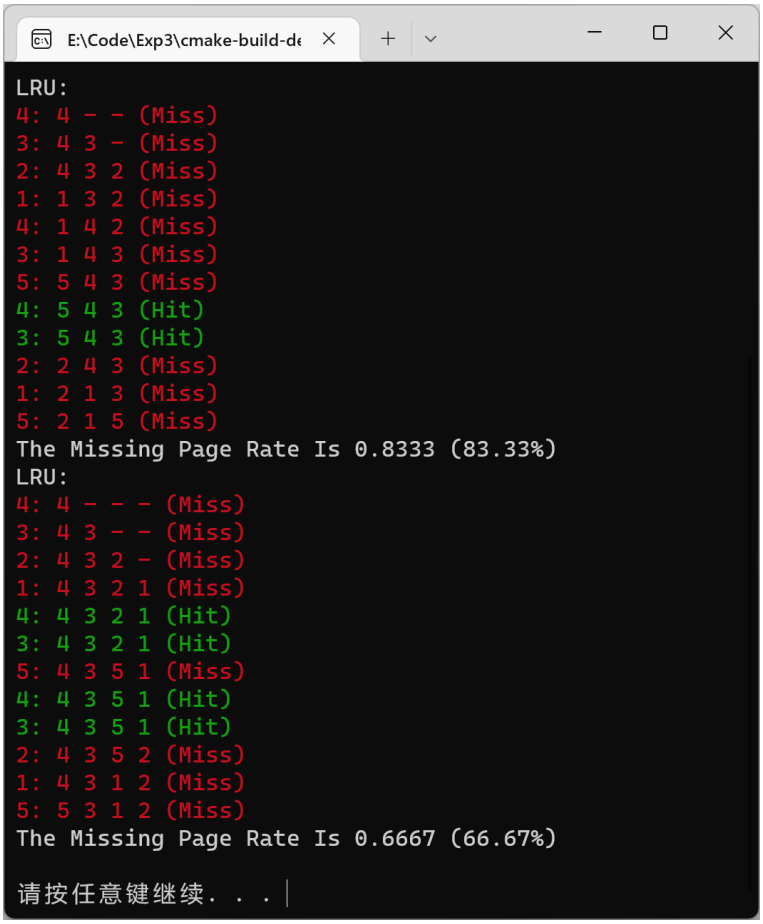
物理块数 m=3

访问页面	4	3	2	1	4	3	5	4	3	2	1	5
内存块 1	4	4	4	1	1	1	5			5	5	
内存块 2		3	3	3	4	4	4			2	2	
内存块 3			2	2	2	3	3			3	1	
是否缺页	√	√	√	√	√	√	√			√	√	

物理块数 m=4

访问页面	4	3	2	1	4	3	5	4	3	2	1	5
内存块 1	4	4	4	4			5	5	5	5	1	1
内存块 2		3	3	3			3	4	4	4	4	5
内存块 3			2	2			2	2	3	3	3	3
内存块 4				1			1	1	1	2	2	2
是否缺页	√	√	√	√			√	√	√	√	√	√

程序输出如下：



实验体会

本次实验中，通过使用代码模拟实现了三种页面调度置换算法，我对操作系统内部的页面置换方式有了更深刻的认识和感受。OPT 算法是理想型的算法，因为在现实生活中，根本无法知道下一个到来的页面是哪一个，所以只能作为评判页面置换算法优劣的一个标准。LRU 算法的本质则是一种栈的实现。