**虚拟内存：页面置换算法**

**1 实验目的**

请求页式虚拟内存管理是常用的虚拟存储管理方案之一。通过模拟实现请求页式存储管理的几种基本页面置换算法，了解虚拟存储技术的特点，掌握虚拟存储请求页式存储管理中几种基本页面置换算法的基本思想和实现过程，并比较它们的效率。

**2 实验内容**

本实验使用C语言编程模拟一个拥有若干个虚拟页的进程在给定的若干个物理页中运行，并在缺页中断发生时分别使用FIFO、OPT和LRU算法进行页面置换的情况。虽然已经附有参考代码，请同学们完成以下工作：

**（1）按照上课讲述的步骤与格式，计算下面的题目：**

考虑下述页面走向：

1，2，3，4，2，1，5，6，2，1，2，3，7，6，3，2，1，2，3，6

当内存块数量分别为3时（或者说是页框数是3，或者说帧数是3），试问FIFO、LRU、OPT（或者叫MIN）这三种置换算法的缺页次数各是多少，淘汰的页面是哪些？缺页中断率是多少？当内存块数量是4呢？

**（2）查看参考代码，在添加注释理解代码的同时，补充缺失的代码。**

**（3）对代码进行编译，验证与（1）的计算结果是否一致。如果存在问题，是什么问题？你是怎么解决的？**

**3 实验准备**

**关于三种页面置换算法的描述**

1. 先进先出页面置换算法（FIFO）

FIFO算法即淘汰最先进入内存的页面，即选择在内存中驻留时间最久的页面予以淘汰。该算法实现简单只需要把一个进程已调入内存的页面，按先后次序链接成一个队列，并设置一个指针，称为替换指针，使它总是指向最老的页面。

1. 最佳置换算法（OPT）

它是由Belady于1966年提出的一种理论上的算法。其所选择的被淘汰页面，将是以后永不使用的或许是在最长（未来）时间内不再被访问的页面。采用最佳置换算法，通常可保证获得最低的缺页率。但由于无法预知一个进程在内存的若干个页面中，哪一个页面是未来最长时间内不再被访问的，因而该算法是无法实现的，但可以利用此算法来评价其他算法。

1. 最近最久未使用置换算法（LRU）

最近最久未使用（LRU）置换算法，是根据页面调入内存后的使用情况进行决策的。由于无法预测各页面将来的使用情况，只能利用“最近的过去”作为“最近的将来”的近似。因此，LRU置换算法是选择最近最久未使用的页面予以淘汰。该算法赋予每个页面一个访问字段，用来记录一个页面自上次被访问以来所经历的时间t，当须淘汰一个页面时，选择现有页面中其t值最大的，即最近最久未使用的页面予以淘汰。

**4 实验要求**

**（1）可从附件的两个参考程序中选择其一进行补充，也鼓励同学自己实现。**

**（2）必须将代码与实验报告一起打包上传FTP，否则本次实验视为无效。**

**5. 实验结果**

**（1）按照上课讲述的步骤与格式，计算题目。**

**=================== 计算步骤与结果写在下面 ===============**

**（2）对代码进行编译，验证与你的计算结果是否一致。如果存在问题，是什么问题？你是怎么解决的？**

**==================== 存在问题及原因 ====================**

**==================== 解决方法 =========================**