**存储管理**

一、实验说明

**1. 虚页和实页的表示**

可定义以下描述虚页和实页的结构：

|  |
| --- |
| pn |
| pfn |
| time |

|  |
| --- |
| pn |
| pfn |
| next |

虚页结构 实页结构

在虚页结构中，pn代表虚页号，因为共10个虚页，所以pn的取值范围是0—9。pfn代表实页号，当一虚页未装入实页时，此项值为-1；当该虚页已装入某一实页时，此项值为所装入的实页的实页号pfn。time项在FIFO算法中不使用，在LRU中用来存放对该虚页的最近访问时间。

在实页结构中，pn代表虚页号，表示pn所代表的虚页目前正放在此实页中。pfn代表实页号，取值范围（0～n-1）由动态指派的实页数n所决定。next是一个指向实页结构体的指针，用于多个实页以链表形式组织起来，关于实页链表的组织详见下面第4点。

**2．关于缺页次数的统计**

为计算命中率，需要统计在20次的虚页访问中命中的次数。为此，程序应设置一个计数器count，来统计虚页命中发生的次数。每当所访问的虚页的pfn项值不为-1，表示此虚页已被装入某实页内，此虚页被命中，count加1。最终命中率=count/20\*100%。

**3．LRU算法中“最近最久未用”页面的确定**

为了能找到“最近最久未用”的虚页面，程序中可引入一个时间计数器countime，每当要访问一个虚页面时，countime的值加1，然后将所要访问的虚页的time值设置为增值后的当前countime值，表示该虚页的最后一次被访问时间。当LRU算法需要置换时，从所有已分配实页的虚页中找出time值最小的虚页就是“最近最久未用”的虚页面，应该将它置换出去。

**4．算法中实页的组织**

因为能分配的实页数n是在程序运行时由用户动态指派的，所以应使用链表组织动态产生的多个实页。为了调度算法实现的方便，可以考虑引入free和busy两个链表：free链表用于组织未分配出去的实页，首指针为free\_head，初始时n个实页都处于free链表中；busy链表用于组织已分配出去的实页，首指针为busy\_head，尾指针为busy\_tail，初始值都为null。当所要访问的一个虚页不在实页中时，将产生缺页中断。此时若free链表不为空，就取下链表首指针所指的实页，并分配给该虚页。若free链表为空，则说明n个实页已全部分配出去，此时应进行页面置换：对于FIFO算法要将busy\_head 所指的实页从busy链表中取下，分配给该虚页，然后再将该实页插入到busy链表尾部；对于LRU算法则要从所有已分配实页的虚页中找出time值为最小的虚页，将该虚页从装载它的那个实页中置换出去，并在该实页中装入当前正要访问的虚页。

二、程序流程图

1.FIFO算法



2.LRU算法

