**操**

**作**

**系**

**统**

**上**

**机**

**报**

**告**

**五**

班级：19计科\*班（本）

学号：

姓名：

周次：

上机时间：

《操作系统》实验(五)

1. **实验目的**
   1. 模拟页面置换算法FIFO;
   2. 模拟页面置换算法LRU；

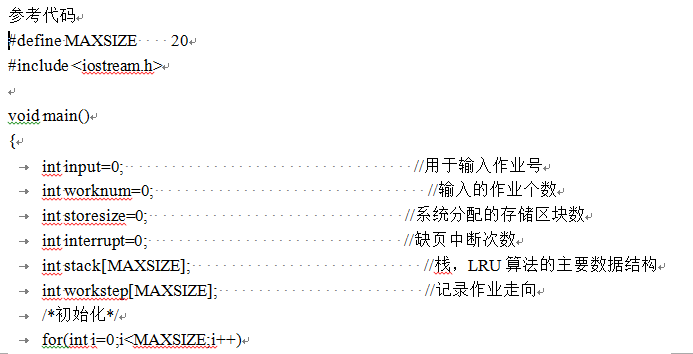
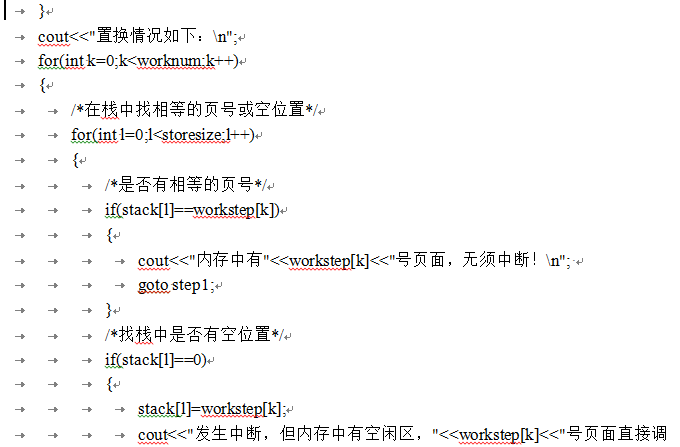
加深对FIFO算法和LRU算法的理解。

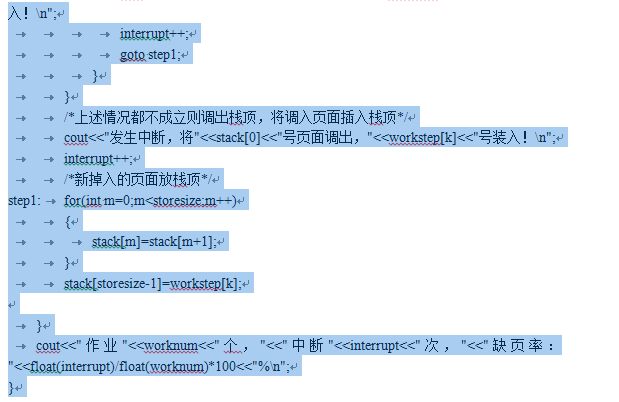
1. **实验内容**
   1. 用C语言模拟实现页面置换算法FIFO;
   2. 用C语言模拟实现页面置换算法LRU；
2. **实验准备**

**实验原理**

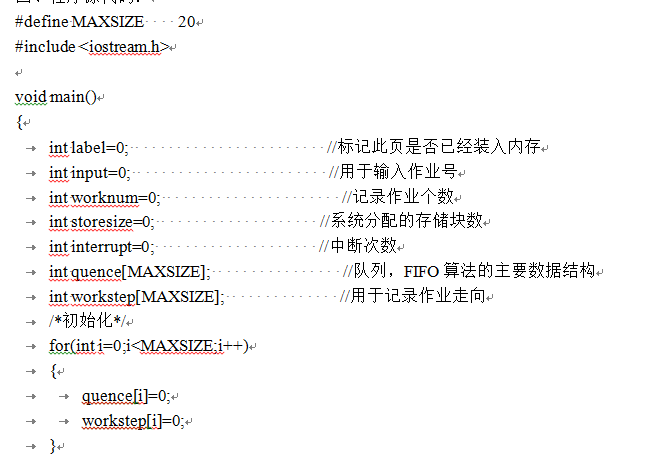
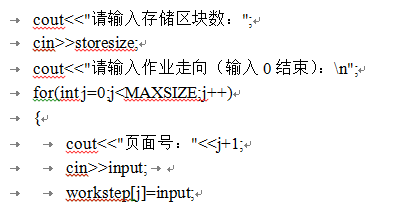
1. 要求模拟FIFO算法，一个作业有多少个进程，处理机只分配固定的主存页面供该作业执行。往往页面数小于进程数，当请求调页程序调进一个页面时，可能碰到主存中并没有空闲块的情况，此时就产生了在主存中淘汰哪个页面的情况。此算法的实质是，总是选择在主存中居留最长时间的页面淘汰。理由是：最早调入主存的页，其不再被访问的可能性最大。
2. 要求模拟LRU算法，其基本原理为：如果某一个页面被访问了，它很可能还要被访问；相反，如果它长时间不被访问，再最近未来是不大可能被访问的。
3. **参考代码**

**LRU**

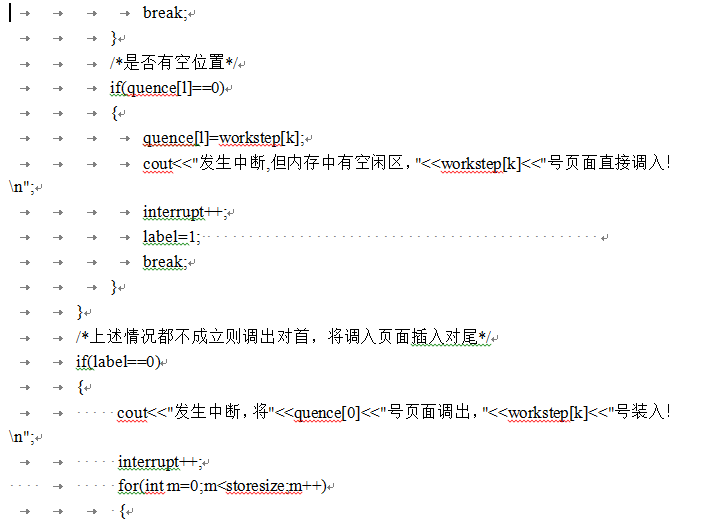


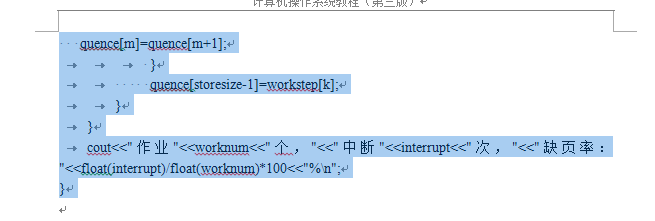
FIFO

  if(input==0)

{

cout<<"输入结束！\n";



1. **程序调试（此为模板，具体调试选另外的例子）**

**LRU**

调试示例1：**（再验证此章作业习题，4个物理块的情况）**

请输入存储区块数：3

请输入作业走向（输入0结束）：

页面号1：4

页面号2：3

页面号3：2

页面号4：1

页面号5：4

页面号6：3

页面号7：5

页面号8：4

页面号9：3

页面号10：2

页面号11：1

页面号12：5

页面号13：0

输入结束！

置换情况如下：

发生中断，但内存中有空闲区，4号页面直接调入！

发生中断，但内存中有空闲区，3号页面直接调入！

发生中断，但内存中有空闲区，2号页面直接调入！

发生中断，将4号页面调出，1号装入！

发生中断，将3号页面调出，4号装入！

发生中断，将2号页面调出，3号装入！

发生中断，将1号页面调出，5号装入！

内存中有4号页面，无须中断！

内存中有3号页面，无须中断！

发生中断，将5号页面调出，2号装入！

发生中断，将4号页面调出，1号装入！

发生中断，将3号页面调出，5号装入！

作业12个，中断10次，缺页率：83.3333%

Press any key to continue

FIFO

调试2：（再验证课后习题）

请输入存储区块数：4

请输入作业走向（输入0结束）：

页面号1：4

页面号2：3

页面号3：2

页面号4：1

页面号5：4

页面号6：3

页面号7：5

页面号8：4

页面号9：3

页面号10：2

页面号11：1

页面号12：5

页面号13：0

输入结束！

置换情况如下：

发生中断,但内存中有空闲区，4号页面直接调入！

发生中断,但内存中有空闲区，3号页面直接调入！

发生中断,但内存中有空闲区，2号页面直接调入！

发生中断,但内存中有空闲区，1号页面直接调入！

内存中有4号页面，无须中断！

内存中有3号页面，无须中断！

发生中断，将4号页面调出，5号装入！

发生中断，将3号页面调出，4号装入！

发生中断，将2号页面调出，3号装入！

发生中断，将1号页面调出，2号装入！

发生中断，将5号页面调出，1号装入！

发生中断，将4号页面调出，5号装入！

作业12个，中断10次，缺页率：83.3333%

Press any key to continue

1. **实验心得**