《操作系统原理》实验报告

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 姓名 | 刘逸凡 | 学号 | U202012155 | 专业班级 | 网安2004 | 时间 | 2022.12.13 |

**一、实验目的**

1）理解页面淘汰算法原理，编写程序演示页面淘汰算法。

2）验证Linux虚拟地址转化为物理地址的机制

3）理解和验证缺页处理的流程。

4）理解设备是文件的概念。

5）掌握Linux模块、驱动的概念和编程流程

**二、实验内容**

1）Windows/Linux模拟实现FIFO或LRU页面淘汰算法。

2）Linux下利用/proc/pid/pagemap技术计算某个变量或函数虚拟地址对

应的物理地址等信息。

3）研读并修改Linux内核的缺页处理函数do\_no\_page 或页框分配函数

get\_free\_page，并用printk打印调试信息。注意：需要编译内核。。

4）编写一个Linux内核模块，并完成模块的安装/卸载等操作。

5）编写Linux驱动程序（字符类型或杂项类型）并编程应用程序测试。

功能：write几个整数进去，read出其和或差或最大值。

6）编写Linux驱动程序（字符类型或杂项类型）并编程应用程序测试。

功能：有序读和写内核缓冲区，可以重复读，可以覆盖写。返回实际读写

字节数

**三、实验过程**

**3.1 模拟实现页面淘汰算法**

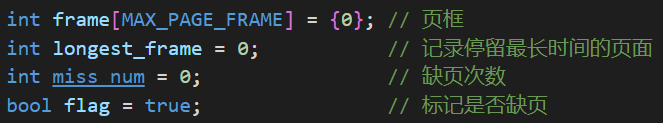
1. FIFO算法

FIFO淘汰停留时间最长的页面，实际上即为先进先出的算法，在遍历页面序列的过程中，进行类似队列的循环淘汰的算法

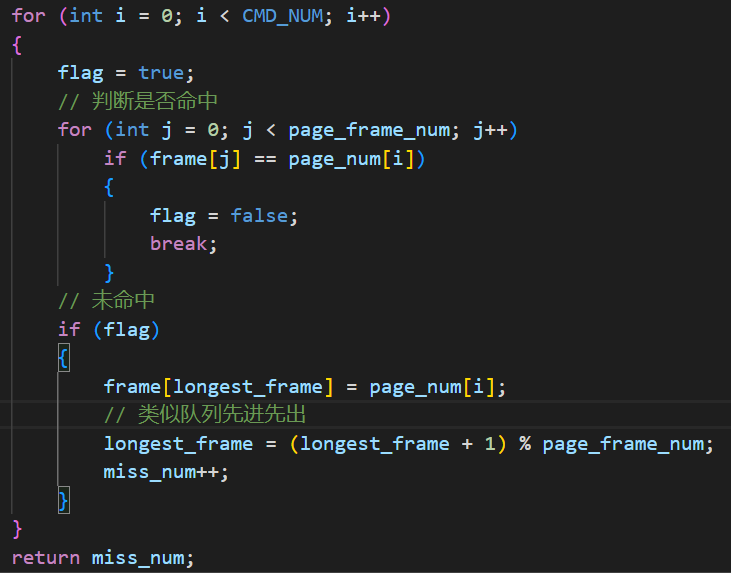
定义全局变量如下所示



函数变量如下图所示



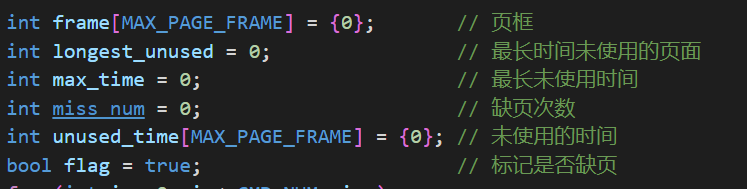
主要遍历过程如下所示



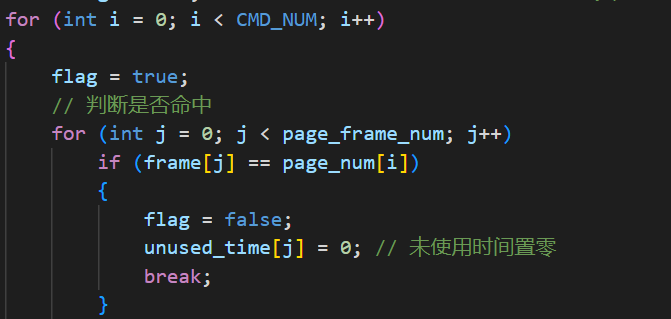
返回缺页的次数

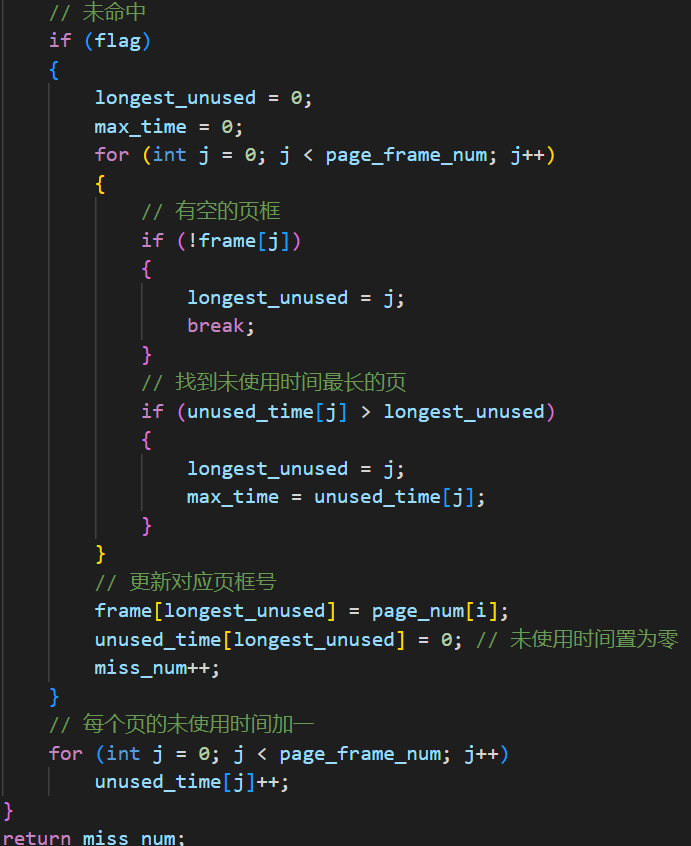
1. LRU算法

LRU算法每次淘汰未使用时间最长的页面，所以可以记录每个页面距离上一次使用的时间，每次淘汰时淘汰时间最长的，并更新每个的时间，函数变量如下图所示



遍历过程如下图所示

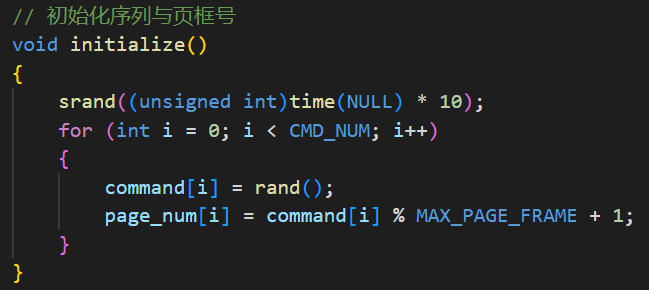




最后同样返回缺页次数

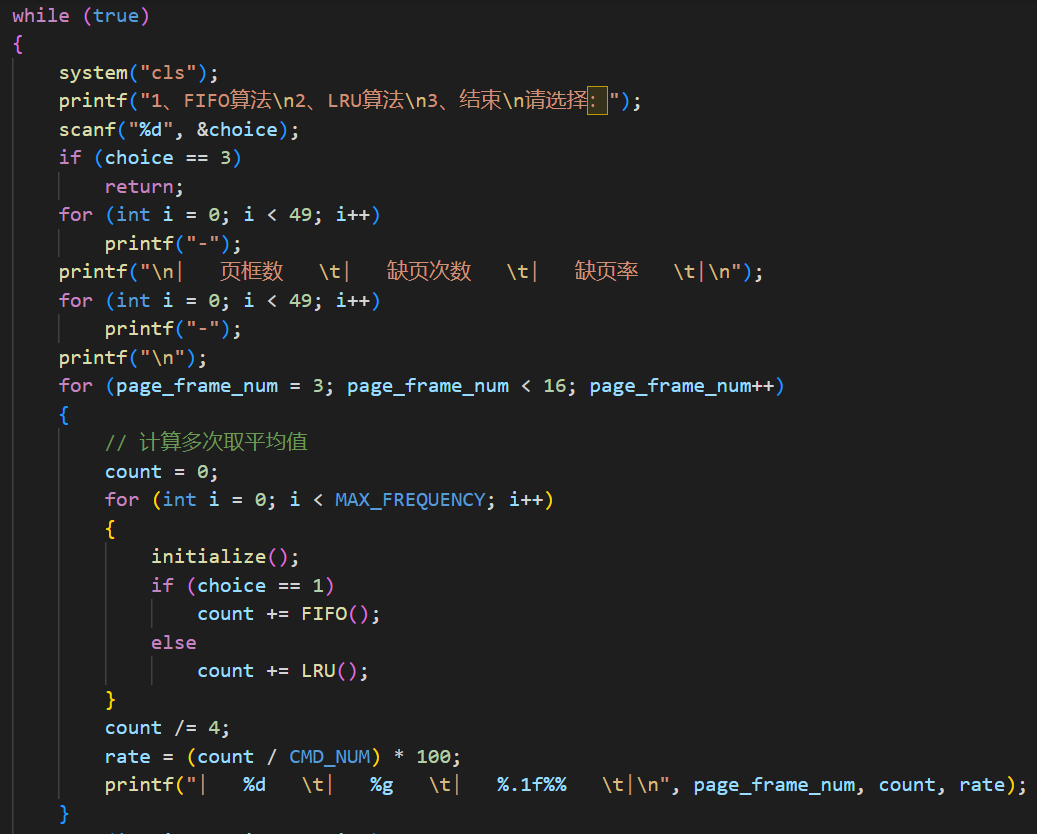
3）初始化页面序列

为了提高准确率，除了每个页面都使用随机数生成以外，每次随机数的种子由当前时间生成，代码如下图所示



4）主程序

每次运行算法时会执行多次取平均值，同时改变页框数以观察页框数与缺页率的关系，主要代码如图所示

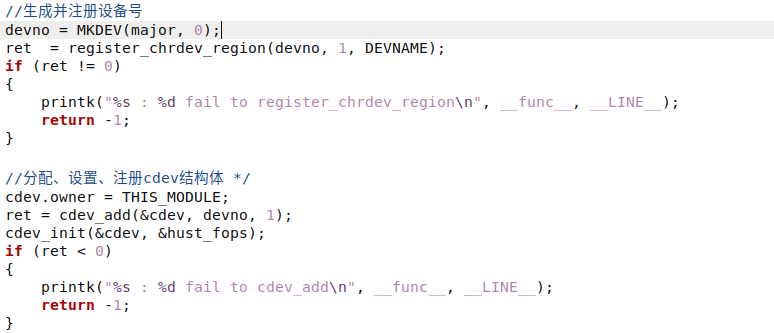


**3.2 编写linux驱动程序**

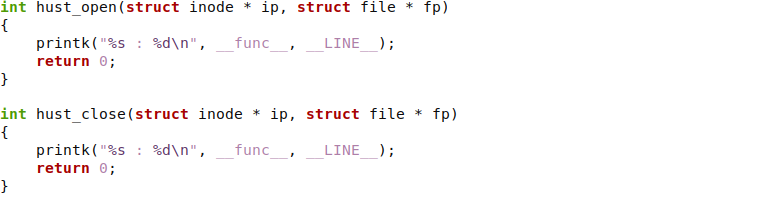
1）驱动程序编写

在驱动程序框架中，含有open、close、read、write几个函数，以及进程的init和exit两个开始、结束函数

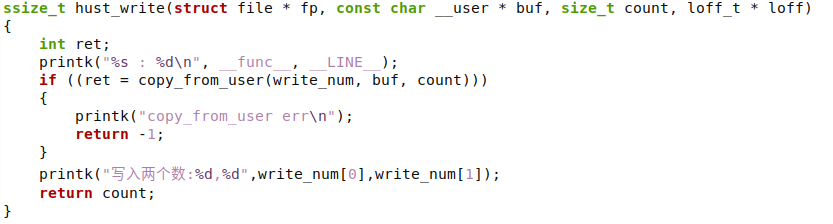
init函数如下所示，完成设备号的注册以及字符设备的cdev结构体的生成



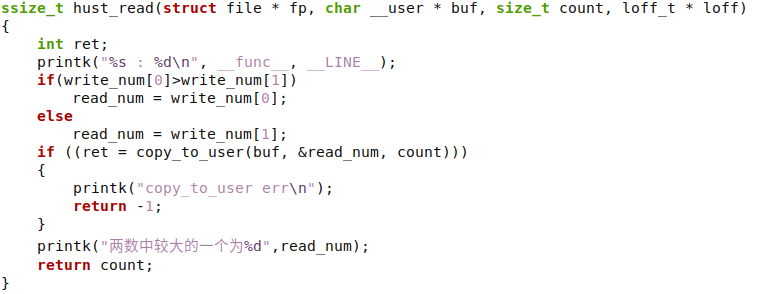
open与close函数为打开与关闭驱动的函数指针，如下所示



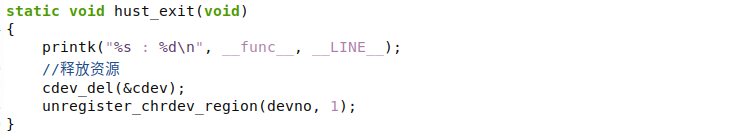
当用户调用write时，将会执行内核中的write函数，从用户中读取数据，如下图所示



当用户调用read时，将会执行内核中的read函数，将数据读入到用户传入的指针中，如图所示

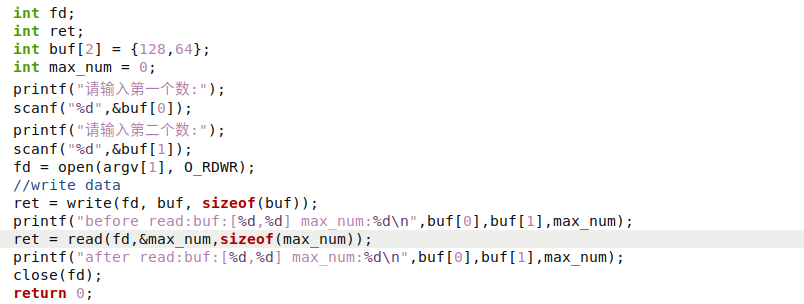


最后有exit函数退出，如下图所示



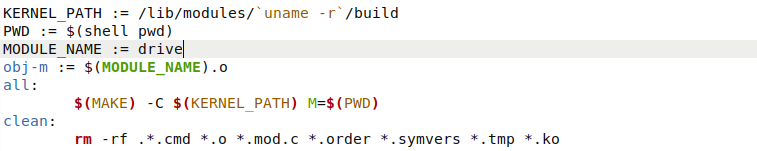
2）测试程序

字符设备驱动将以字符设备文件形式存在，所以通过main函数的参数传入文件路径，对其进行读写，代码如下图所示



3）Makefile文件

通过Makefile生成驱动，如图所示



4）安装驱动

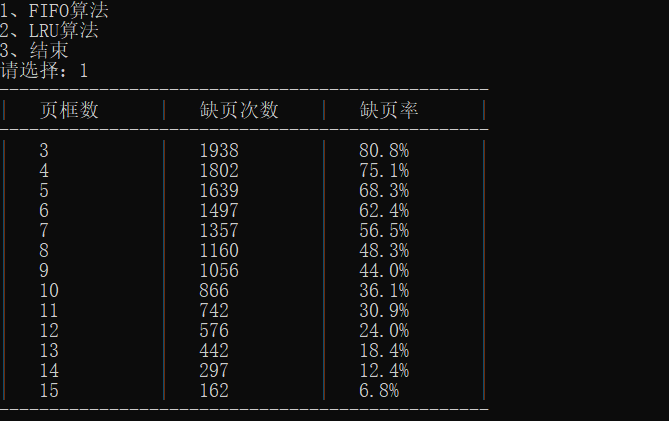
安装驱动并生成测试程序，指令如图所示



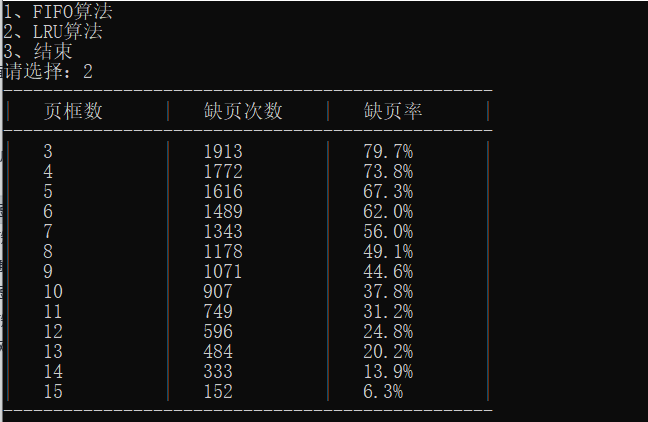
**四、实验结果**：

**4.1 模拟实现页面淘汰算法**

选择执行FIFO算法如图所示

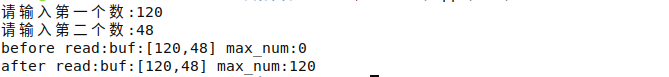


执行LRU算法如下图所示

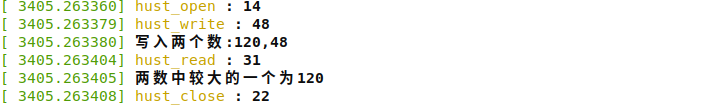


**4.2 编写linux驱动程序**

运行测试程序，可以看到驱动的功能成功被执行，如图所示



执行dmesg指令，查看内核调用，如下图所示



**五、实验错误排查和解决方法**

**4.1 模拟实现页面淘汰算法**

1）FIFO算法的实现中，关于停留时间最长的模拟，最开始也是维护了一个数组来保存每一个页面的停留时间，后来发现实际上就是先进先出，所以每次淘汰直接对页框数加一取模就行了

2）一个不算问题的问题，很少使用vscode，结果这次用的时候发现vscode无法识别”bool”、”true”等符号，需包含<stdbool>才行

**4.2 编写linux驱动程序**

1）刚开始的时候这个实验真的无从下手，网上查阅了相关资料才大概了解驱动具体的框架

2）编写makefile文件的时候，生成的目标文件的名字一定要和源代码文件名字相同

**六、实验参考资料和网址**

**（1）**[C语言实现FIFO，LRU淘汰算法\_AspirinK的博客-CSDN博客\_lru页面淘汰算法c语言](https://blog.csdn.net/qq_45740212/article/details/113366246)

**（2）[（三）写一个完整的Linux驱动程序访问硬件并写应用程序进行测试\_hanp\_linux的博客-CSDN博客](https://blog.csdn.net/hanp_linux/article/details/90474432)**