**同济大学计算机系**

**操作系统实验报告**

****

**学 号 2152809**

**姓 名 曾崇然**

**专 业 计算机科学与技术**

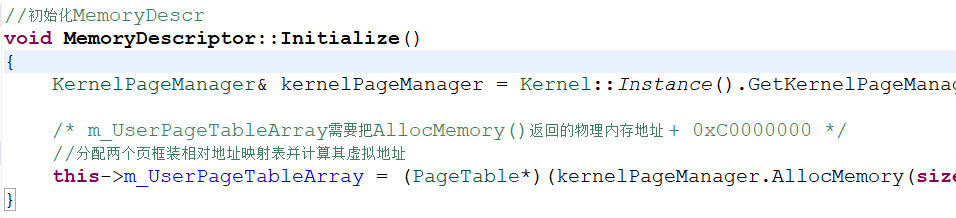
**授课老师 邓蓉老师**

**完成内容：完成了实验步骤1和2，即读通了程序bing 完成了将相对虚实地址映射表赋值为NULL并跑通**

**一. 读子程序并进行注释**

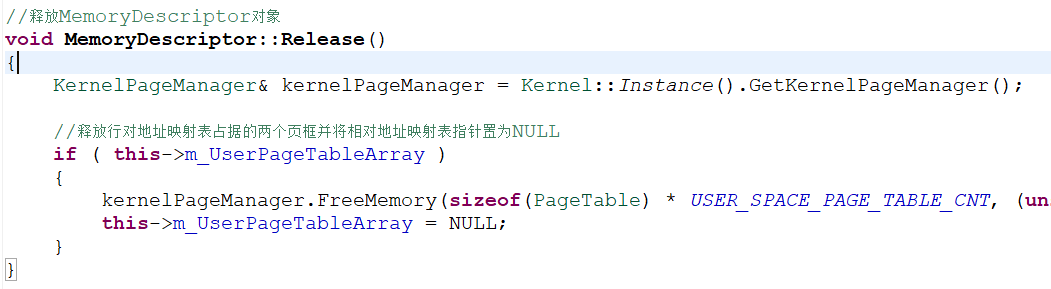
**1. Initialize:**

初始化MemoryDescriptor



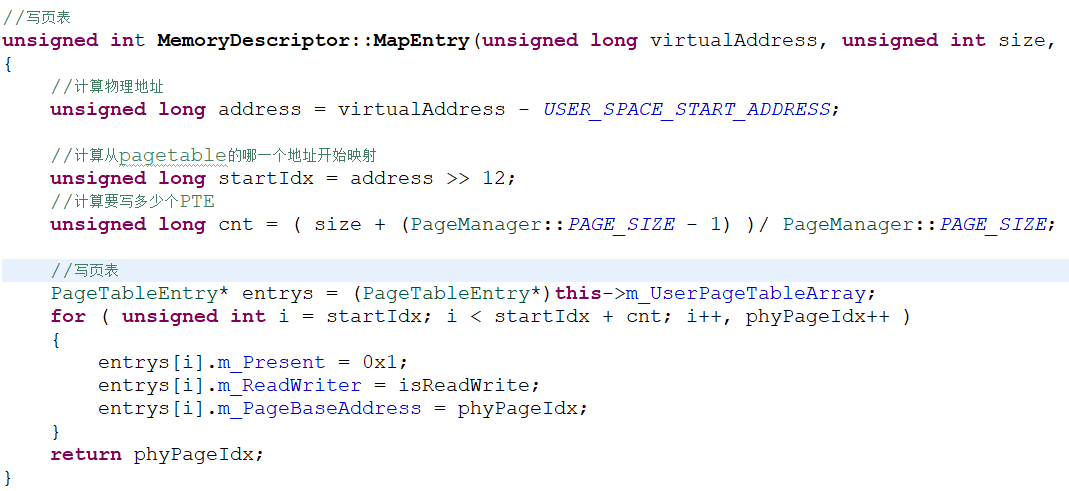
**2. Release:**

释放MemoryDescriptor对象



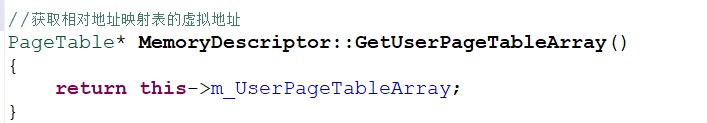
**3. MapEntry:**

写页表



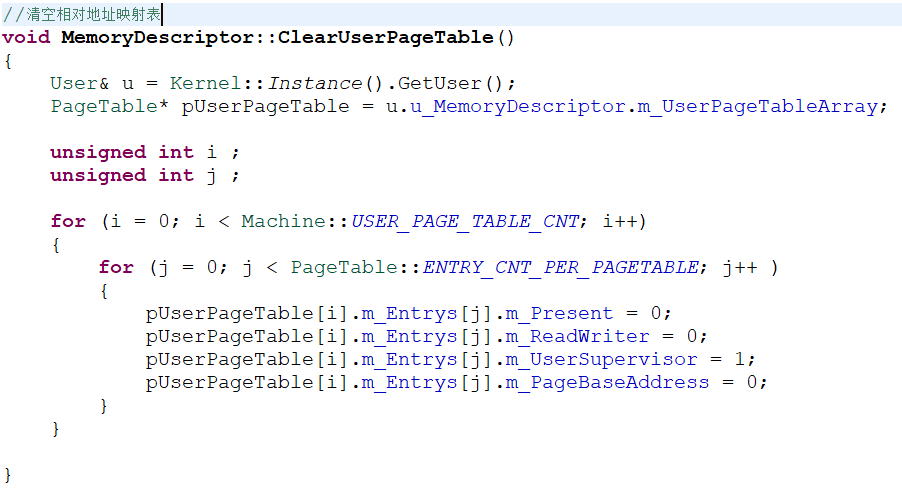
**4. GetUserPageTableArray:**

获取相对地址映射表的虚拟地址



**5. ClearUserPageTable:**

清空相对地址映射表



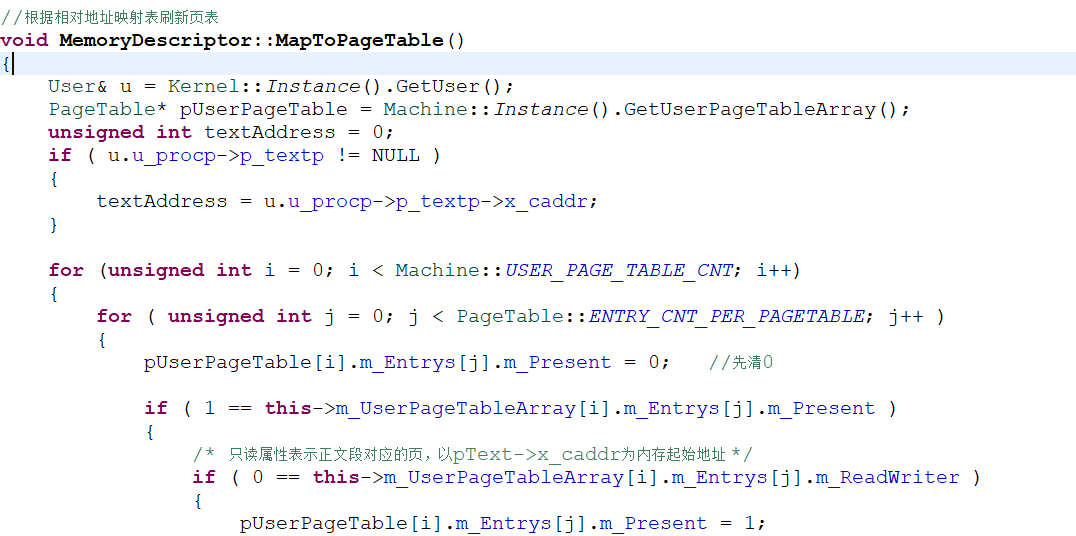
**6. EstablishUserPageTable:**

建立相对地址映射表



**7. MapToPageTable:**

根据相对地址映射表刷新页表



**二. 去除相对地址映射表的重要性**

**1. 这个数据结构是没有其必要性的：**相对虚实地址映射表能够用来写系统页表，但是通过各段的长度和p\_addr，p\_text等信息就能够直接写出系统页表，因此相对虚实地址映射表是冗余的

**2. 占用太多核心空间：**一个进程需要一张相对虚实地址映射表，一张相对虚实地址映射表占用两个页框大小的核心空间，这徒劳的增加了系统核心空间的开销

综上，去除相对虚实地址映射表不仅不会影响系统的功能，反而能使系统更加的合理和高效。

**三. 将相对地址映射表赋值为NULL并跑通程序**

**1. initialize:**

将相对虚实地址映射表赋值为NULL

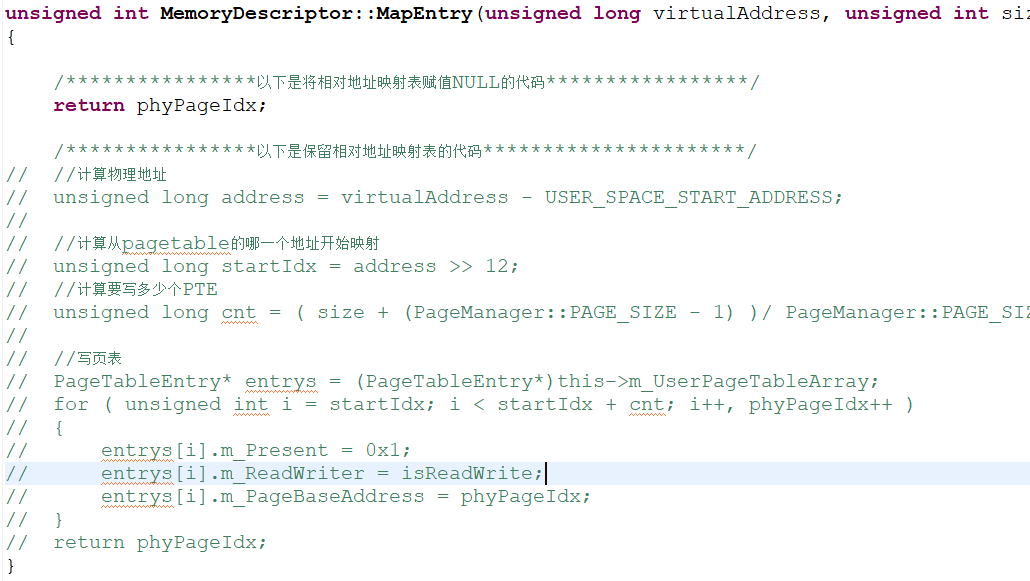


**2. Release:**

因为相对虚实地址映射表已经为NULL了，所以该函数不做操作，无需修改

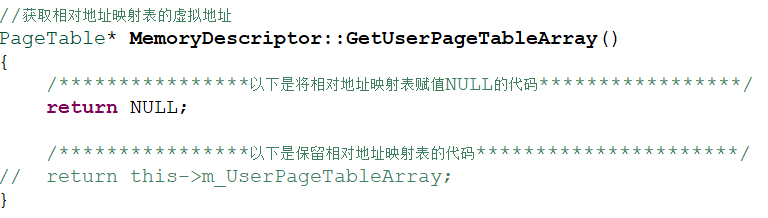
**3. MapEntry:**

在建立相对虚实地址表的函数中调用，现在不使用相对地址映射表，所以将其内容注释掉即可



**4. GetUserPageTableArray:**

现在相对地址映射表为NULL，所以直接返回NULL即可



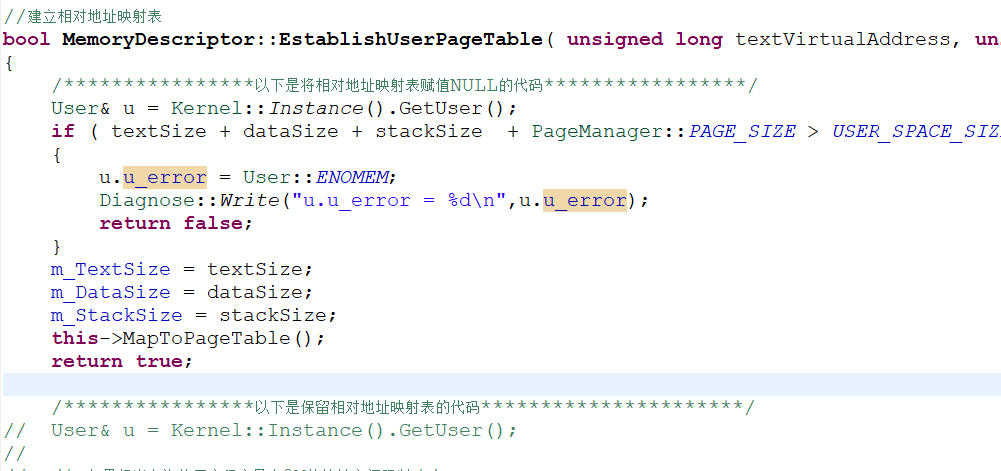
**5. ClearUserPageTable:**

因为拿掉了相对虚实地址映射表，所以清除时无需操作，全部注释掉即可



**6. EstablishUserPageTable:**

大小超过8M地址空间即报错，小于则为代码段、数据段和堆栈段赋值，并调用MapToPageTable刷新页表

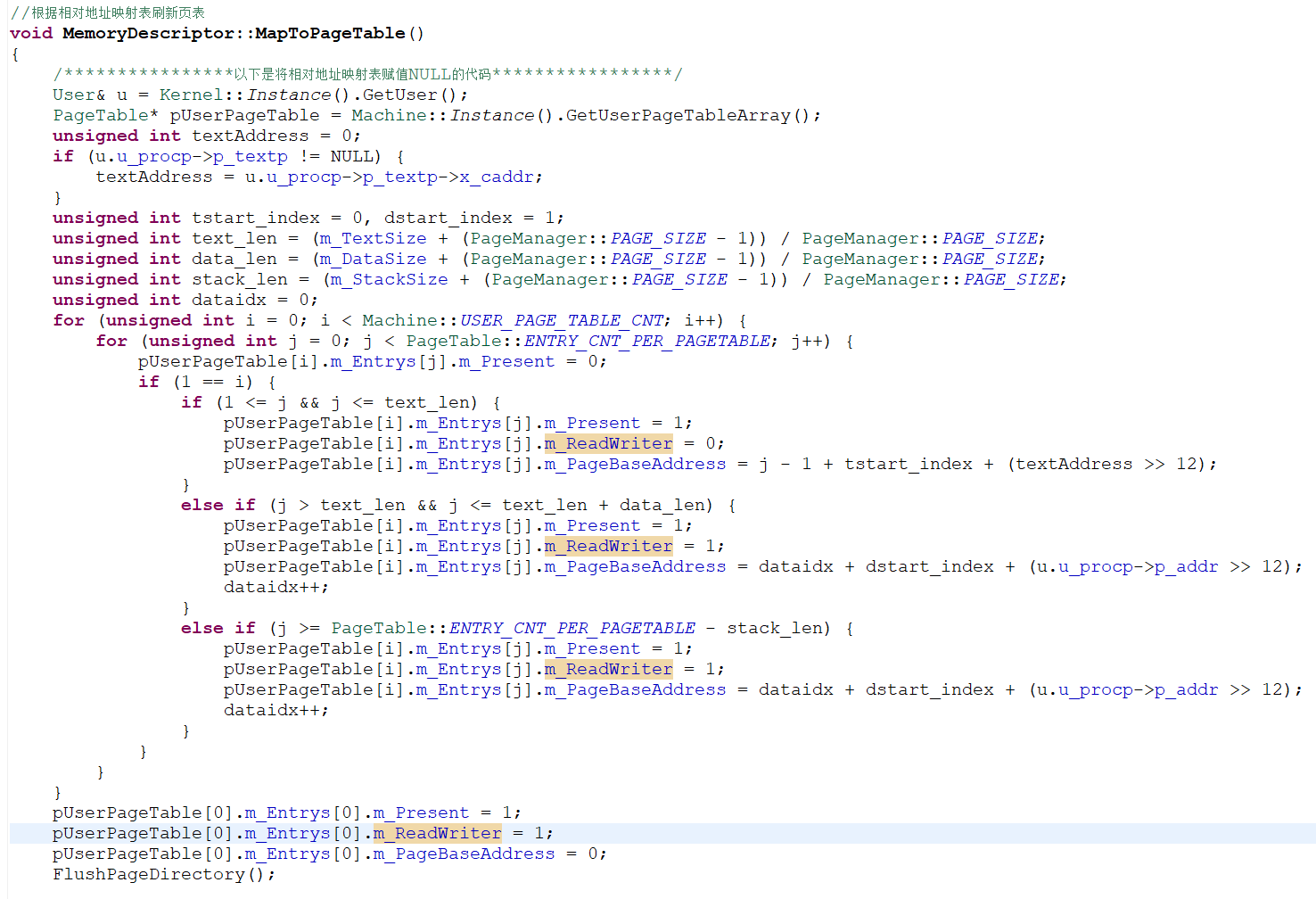


**7. MapToPageTable:**

①计算各段起始偏移量和长度

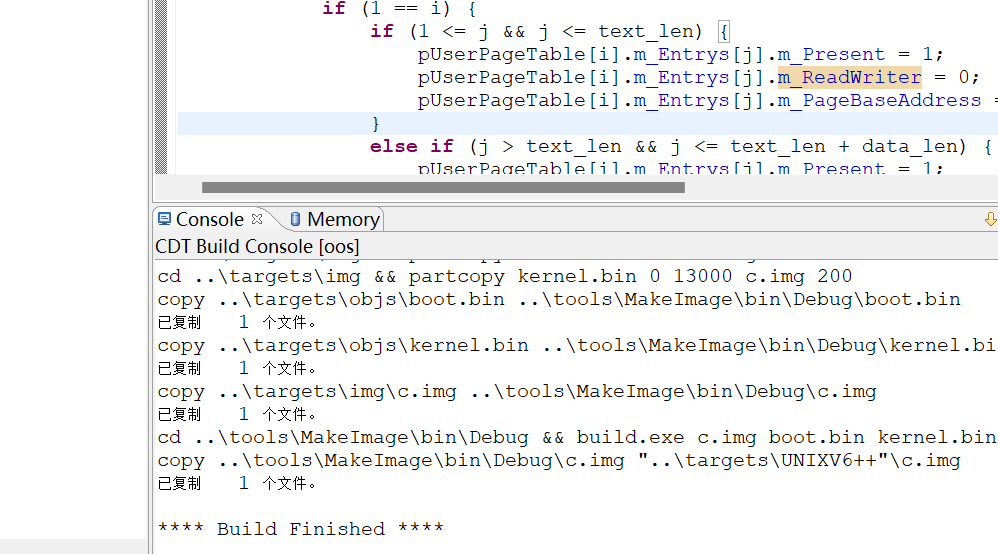
②遍历每张用户页表的每一项进行写入

③根据代码段、数据段、堆栈的不同特性写其p值和w/r值，根据p\_text，p\_addr来计算base以回避掉相对虚实地址映射表



**四. 实验现象**

**1. 编译成功**



**2. 能成功运行**

