**实验报告**

课程名称： 操作系统

任课教师： 何永忠

学生姓名： 战泓夷

学生学号： 16281022

专业班级： 安全1601

学院名称： 计算机与信息技术学院

2019 年 3 月 10 日

### 实验一: 操作系统初步

**一、（系统调用实验）了解系统调用不同的封装形式。**

1. 参考下列网址中的程序。阅读分别运行用API接口函数getpid()直接调用和汇编中断调用两种方式调用Linux操作系统的同一个系统调用getpid的程序(请问getpid的系统调用号是多少？linux系统调用的中断向量号是多少？)。

Getpid获取进程标志号

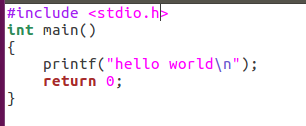


答：getpid的系统调用号是20

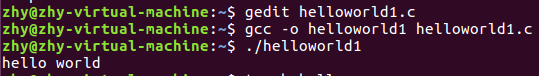
linux系统调用的中断向量号是0x80

1. 上机完成习题1.13
2. 用c语言

程序代码：

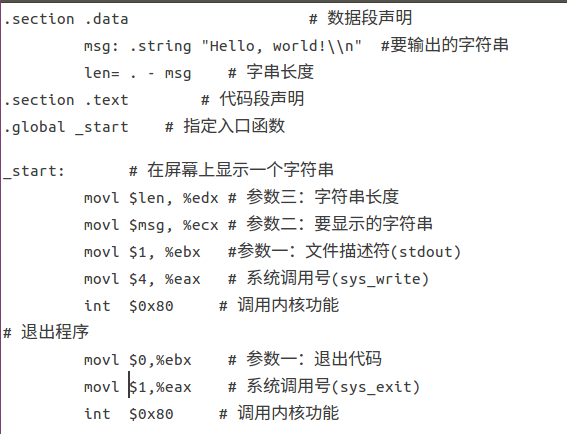


结果：

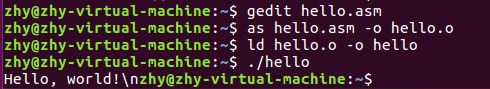


1. 用汇编语言

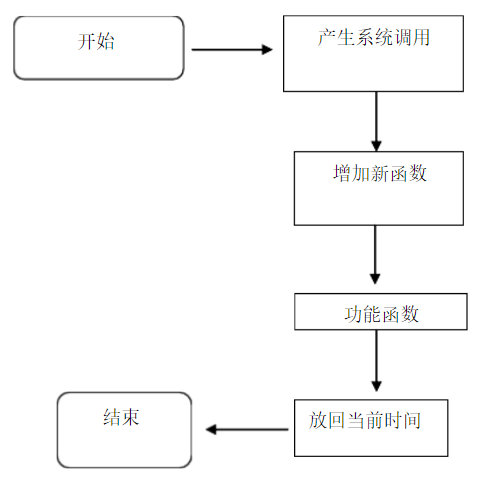
程序代码：



结果：



1. 阅读pintos操作系统源代码，画出系统调用实现的流程图。

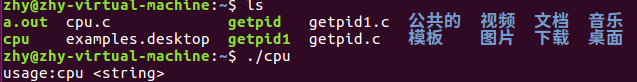


**二、（并发实验）根据以下代码完成下面的实验。**

1. 编译运行该程序（cpu.c），观察输出结果，说明程序功能。

(编译命令： gcc -o cpu cpu.c –Wall)（执行命令：./cpu）

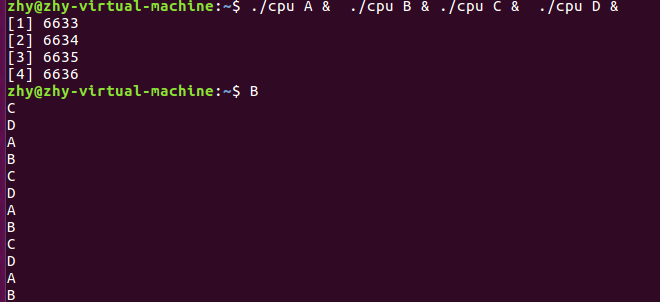
结果：



程序功能：

argc为参数个数， \*argv[]为参数，当argc等于1或者大于2时，输出usage: cpu <string>，将第二个参数赋值给字符串str（这里的字符相当于进程），用while循环输出字符（sleep(1);就是休眠1秒）

1. 再次按下面的运行并观察结果：执行命令：./cpu A & ; ./cpu B & ; ./cpu C & ; ./cpu D &程序cpu运行了几次？他们运行的顺序有何特点和规律？请结合操作系统的特征进行解释。

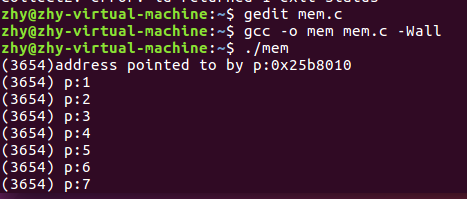
结果：

答：这个程序体现了cpu的并发过程，当程序有ABCD四个进程同时进行时，程序cpu运行了4次，他们的特点和规律：宏观上多个程序在同时进行，但是在微观上这些程序只能是分时的交替进行。体现了并发性的特点。

**三、（内存分配实验）根据以下代码完成实验。**

1.阅读并编译运行该程序(mem.c)，观察输出结果，说明程序功能。(命令： gcc -o mem mem.c –Wall)

结果：



程序功能：

首先给p分配空间，使用断言，当p不为空，则输出p的地址和进程号，给p赋值为0，用while循环加1，并输出当前p的值和进程号。

1. 再次按下面的命令运行并观察结果。两个分别运行的程序分配的内存地址是否相同？是否共享同一块物理内存区域？为什么？命令：./mem &; ./mem &

结果：



答：如图可以看出，两个分别运行的程序分配的内存地址不相同，由于内存地址不同，我认为它们不共享同一块物理内存。

**四、（共享的问题）根据以下代码完成实验。**

1. 阅读并编译运行该程序，观察输出结果，说明程序功能。（编译命令：gcc -o thread thread.c -Wall –pthread）（执行命令1：./thread 1000）

答：worker是一个计数功能的函数，argc为参数个数， \*argv[]为参数，当argc等于1或者大于2时，输出usage: cpu <string>，把命令行参数中文件名后的第二个字符串转化为整数，赋值给loops，创建两个线程，观察操作系统中，两个线程同时运行对共享变量的影响。

1. 尝试其他输入参数并执行，并总结执行结果的有何规律？你能尝试解释它吗？（例如执行命令2：./thread 100000）（或者其他参数。）

答：由图我们可以看出，共享变量的存在，使得每个线程都可以使用它，因此执行的结果是参数的二倍（因为是两个线程）。

1. 提示：哪些变量是各个线程共享的，线程并发执行时访问共享变量会不会导致意想不到的问题。

答：可能会导致，由于每个cpu线程都有一定的时间限制，当达到这个限制时，超过的次数就不会计入，因此导致可能输出的final value比实际只要小。

结果：



1. **实验心得**

通过本次实验，我了解了操作系统给应用成提供服务的方式-系统调用，通过c语言和汇编编程，理解了系统调用与普通函数调用的不同之处，同时通过执行不同的程序，更直观的体会到在操作系统的学习中并发、内存分配、共享的特点，对以后更深入地学习打下坚实的基础。