**暨南大学本科实验报告专用纸**

课程名称 Linux编程 成绩评定

实验项目名称 Linux信号 指导教师 李军

学生姓名 罗杨 学号 2021101353

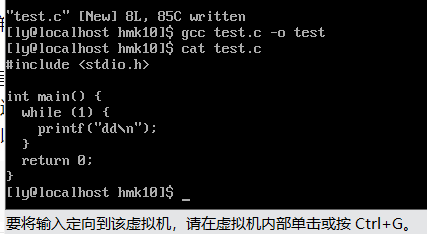
学院 智能科学与工程 系 专业 人工智能

实验时间 2024 年 4 月 28 日 上 午～ 4 月 28 日 下 午

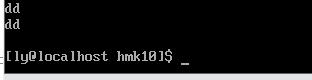
1. 实验内容
   * + 1. SIGINT信号发送

在main函数中，放置一个死循环。编译运行后，按ctrl+c键，测试能否终止程序运行。

编写一个死循环：

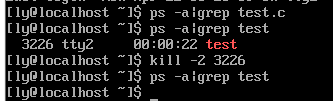


运行并终止：

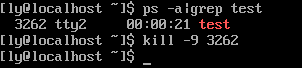


再次运行以上程序，再打开另一个终端，在该终端中，先用ps -a命令查看该程序的pid,然后通过kill 命令向运行中的程序发送SIGINT信号，效果与1中直接按ctrl+c键一样否？

使用ps -a命令可以列出当前终端的所有进程，包括后台运行的进程。找到你的程序的进程ID（PID），然后使用kill命令加上SIGINT信号（或者信号编号2）发送给该PID，可以模拟Ctrl+C的行为。例如，如果PID是1234，那么命令将是kill -2 1234。



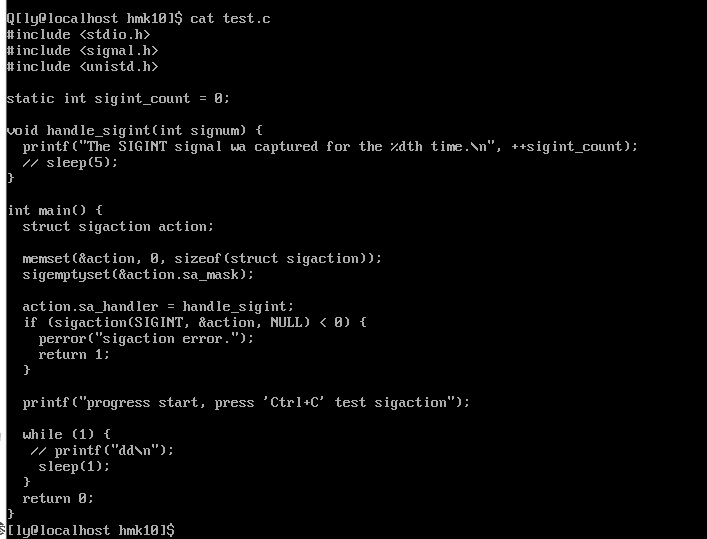
用kill命令发送9号信号给运行中的程序。

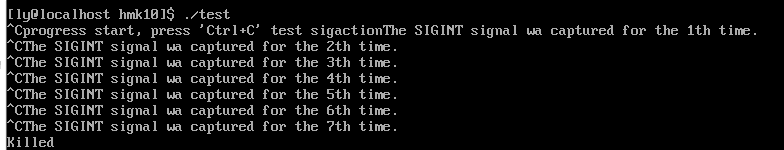


二、捕获SIGINT

1.在程序中用sigaction()函数为SIGINT安装信号处理函数。参考教材P166，但代码的17行是没有必要的。信号处理函数中：

（1）输出你已经第几次捕获了SIGINT信号。应该在信号处理函数里设一个static变量，记录次数。



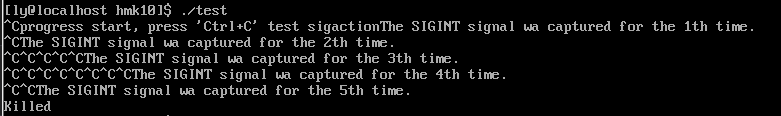
编写程序测试信号处理次数，运行结果如下：  


可以看到键入Ctrl+C后立刻打印捕捉次数。

（2）调用sleep(5)，让信号处理函数模拟经历了较长时间的任务处理。

运行程序后，再用ctrl+c键测试，特别包括连续的几次击键测试“未决”信号，体会信号处理期间，SIGINT已经被屏蔽；当信号处理函数运行完，之前的连续ctrl+c键，只被再递送1次。

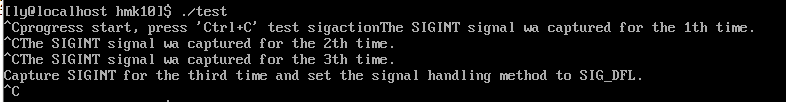
取消sleep(5)的注释，结果如下图：



在键入多个Ctrl+C后只被递送一次。

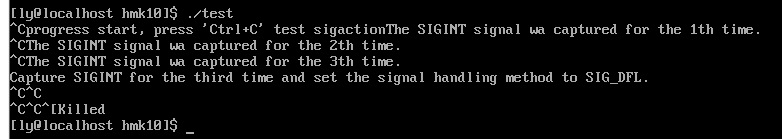
2.在信号处理函数的末尾，判断SIGINT被捕获的次数是否为3，若已到达3次，则用sigaction()函数设置新的信号处理方式分别为SIG\_DFL，SIG\_IGN,参考教材P164倒数第2段关于SIG\_DFL，SIG\_IGN含义的描述。在sigaction结构体的sa\_handler字段应该设置成SIG\_DFL或SIG\_IGN。分别测试它们的效果

设置成SIG\_DFL：



三次后会自动退出程序。

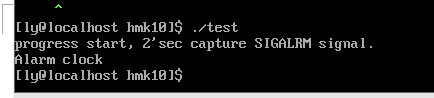
设置成SIG\_IGN：



不能自动退出程序，需要在tty3杀死进程。

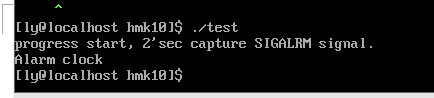
三、定时器

1.在main函数中用alarm函数设置一个2秒的定时器，然后运行一个死循环。观察当定时器到时后会发生什么。



两秒后程序会终止。

2.sigaction()函数设置捕获SIGALRM信号，在信号处理函数中输出一个字符串。



3.用setitimer函数代替alarm函数，设置一个周期性的1秒定时器。在其信号处理函数中，记录触发的次数，当触发了3次后，再次调用setitimer函数，设置定时时间为0，以便取消该定时器。

提示：设定结构体itimerval的值时，可以简洁点。

struct itimerval it={{1,0},{1,0}};

