# 中断与调度部分的复习题

同济大学计算机系 操作系统作业 2023-11-20

学号 姓名

## 一、判断题

1、进程不执行系统调用就不会入睡。

2、现运行进程不响应中断就不会被剥夺。

3、现运行进程不响应中断就不会让出CPU。

4、现运行进程让出CPU后，一定是优先级最高的进程上台运行。

5、Unix V6++系统使用的调度算法是时间片轮转调度。

6、没有中断就没有调度（现运行进程就不会让出CPU）。

7、用户态进程，系统中至多只有一个。

8、Unix V6++，核心态不调度。所以，如果不是入睡或终止，现运行进程不会让出CPU。

二、系统调用不同于一般的子程序调用。请问：UNIX V6++和Linux的系统调用如何

1、传递应用程序想要执行的系统调用号？

2、传递系统调用的参数？

3、将系统调用的返回结果传给应用程序？

三、言简意赅

1、描述20#系统调用的执行过程。

2、描述为Unix V6++系统添加一个新的系统调用的过程。

四、请回答以下问题，言简意赅补齐系统中断响应和调度过程。

1、T0时刻整数秒，系统中SRUN进程PA和PB。现运行进程 PA执行sleep（10）系统调用。

2、现运行进程PA SRUN，正在执行系统调用。T1时刻，响应中断，唤醒一个睡眠进程PB。问，PB进程何时上台运行？简述系统中断响应，调度过程和PB唤醒后上台运行

3、T2时刻整数秒，CPU关中断执行硬盘中断处理程序，硬盘中断处理程序的先前态是用户态。时钟中断何时响应？时钟time的调整是否会延迟，延迟到什么时候？

五、擦掉红色的判断，Unix V6++系统的钟就不走了。为什么 ？ （这个题写着玩）

**void Time::Clock( struct pt\_regs\* regs, struct pt\_context\* context )**

{

……

if( **current->p\_stat == Process::SRUN** &&

(context->xcs & USER\_MODE) == KERNEL\_MODE )

{

发EOI命令；

return;

}

Time::lbolt -= HZ;

Time::time++; //修改wall clock time

……

}

六、Setpri( )有没有一点儿怪。说出你的疑惑，尝试解释原系统设计的合理性，或对它进行质疑。 （这个题写着玩）

七、你自己的任何设计或想法 （这个题写着玩）