

# 中断与调用部分复习题

现运行进程是系统中唯一的用户态进程

- 一: 1. ~~✓~~ 2. ~~✗~~ 3. ~~✓~~ 4. ~~✗~~ 5. ~~✓~~ 6. ~~✗~~ 7. ~~✗~~ 8. ~~✗~~ ✓

如缺中断等

入睡眠终止也会 可剥夺的动态优先级调度

二: 1. 通过 EAX 寄存器 ✓

2. 通过将参数放入寄存器 EBX, ECX, EDX, ESI 和 EDI 中 ✓

3. 通过 EAX 寄存器 ✗ 其它数据可以存入其它通用寄存器, 或者子函数传入的局部变量所有的用户空间内存区域

三: ① 应用程序调用系统调用的子函数

1. ② 系统调用号 20 送入 EAX 寄存器, INT 80 进入核心态 ③ 调用

系统调用入口程序, 完成现场保护 ④ 根据系统调用号

查系统调用子程序入口表, 得到入口地址后执行系统调用

处理程序 ⑤ 该程序将 pid 放入 EAX 中返回用户态 ⑥ 子函数返回

其返回应用程序

系统调用号 20 入核心栈

2. ① 更新系统调用入口表 ✓

② 更新系统调用头文件 ✓

③ 更新系统调用实现文件 ✓

④ 更新用户空间库 ✓

⑤ 重新编译、生成内核和 Unix V6++ 的静态库

四:

1. 系统响应 PA 的中断, p-stat = SWAIT, p-uchan = &start,

① PA 执行系统调用, 陷入核心态, 保护用户态现场, 唤醒 PA, PA 的 p-stat 改为 SRUN, p-uchan = 0, 重算 PA 优先级, 进行调度

② 执行处理函数, 设置 start, 入睡眠并放弃 CPU

③ PB 被选中并 2. 系统响应中断, PB 变为就绪态, 重算其优先级, 若 PB 优先级高于 PA, RUNRUN++, 但是当前为核心态, 直到系统调用中断返回再进行调用调度 (3. PB)

上方执行

3. CPU关中断时时钟中断被屏蔽，中断处理完后开中断，  
允许响应中断，  
会延迟，中断屏蔽可能导致错过原定时钟中断

五：删掉后没有了是否为就绪态的判断，只要是核心态就发送EOI中断结束命令，直接返回了，不执行后续，所以时钟停了 删掉红色的判断没办法调整  $160\mu s$  和  $time$

六：当优先级出现大于255的情况统一为255，会对一些进程的优先级判断产生影响吗 RunRun可能错置

七：是否可以使用float类型来对优先级进行映射以达到更精确的区分呢？