

E10: UNIX V6++进程的睡眠唤醒与图像交换

参考答案与说明

1. 【参考答案】

(1) $RunRun=1$, 为强制调度标志, 表示系统里现在出现了比现运行进程更适合占用处理机的进程。在下次有进程做例行调度时 (一次先前态为用户态的中断返回, 检查 $RunRun$), 将会调用 $Swch$, 实施一次进程调度。

(2) $RunIn=1$, 表示盘交换区上有进程需要进入内存, 但是内存没有足够的空间可供调入, 且寻找不到可供换出的进程图像。此时, 0#进程因为 $RunIn$ 进入睡眠状态。未来, 如果内存出现可供换出的进程 (有进程进入低睡状态或 1 秒计时到), 0#进程被唤醒, 重新寻找可供换出的进程图像。

(3) $RunOut=1$, 表示盘交换区上没有就绪状态的进程需要进入内存。此时, 0#进程因为 $RunOut$ 进入睡眠状态。未来, 如果有一个盘交换区上睡眠的进程被唤醒, 将同时唤醒 0#进程, 醒来的 0#进程上台后, 将该进程的图像换入内存。

2. 【参考答案】

(1) T_0 时刻, 现运行进程 pa 执行 $read$ 系统调用: 进程 pa 由于执行磁盘 I/O, 调用内核函数 $Sleep$, 进入高睡状态, 放弃处理器。 $Sleep$ 中调用 $Swch$, 选中内存中的就绪进程 pb , 使其占用处理器继续执行。

(2) T_1 时刻, pa 启动的 I/O 操作完成: pa 启动的 I/O 操作完成, 正在 CPU 上运行的进程 pb 响应中断请求。在磁盘中断处理中, 唤醒进程 pa 。进程 pa 进入就绪状态, 等待下一次被调度到后上台执行。

(3) T_2 时刻, pa 正在执行时, pc 等待的 I/O 操作完成, 因为此时盘交换区只有一个低睡进程, 所以 0#进程一定因为 $RunOut$ 在睡眠:

- pa 响应中断, 唤醒 pc 进程, $SetRun$ 函数中唤醒因为 $RunOut$ 睡眠的 0#进程;
- 下一次 $Swch$ 时, 0#进程上台执行 $Sched$, 找到进程 pc , 为其申请内存;
- 如果申请成功, 将 pc 进程的图像调入, 释放盘交换区空间, 修改 p_addr , 设置 $SLOAD$ 标志;
- 如果申请不成功, 0#进程选择可以换出的进程;
- 如果找到可以换出的进程 (pa 或 pb), 为该进程申请盘交换空间, 调出, 释放内存, 将 pc 进程的图像调入 (如果换出一个不够, 有可能换出 2 个进程);
- 如果找不到可以换出的进程, 0#进程因为 $RunIn$ 睡眠, 未来有进程进入低睡或 1 秒计时到, 将 0#进程唤醒, 再次尝试先换出, 再换入;
- 何时进程 pc 进入内存, 将等待 $Swch$ 的调度。