

# 2018 操作系统实验（四）

本次实验重点在于掌握：进程的概念、操作系统的系统调用、PV 操作以及进程调度的实现。

## 1. 实现进程调度

参考《Orange's》，在之前搭建的 nasm+bochs 实验平台上实现特定进程调度问题的模拟，具体要求如下。

- 添加一个系统调用 `sys_process_sleep`，其功能是接受一个 `int` 型参数 `milli_seconds`，调用此系统调用的进程会在 `milli_seconds` 毫秒内不被分配时间片。
  - 第六章代码已经有 `sys_get_ticks` 系统调用和基于此的 `mills_delay` 函数，似乎已经有了 `sleep` 的功能，但它本质上还是为进程分配了时间片的，只不过分配的时间片在 `mills_delay` 函数中什么也没做。**本次实验要求的 `sys_process_sleep` 是不分配时间片的。**
- 添加一个系统调用 `sys_disp_str`，其功能接受一个 `char *str` 参数，打印出字符串。
  - 第六章代码已经在 `kliba.asm` 文件中有了 `disp_str` 函数显示字符串，但这是内核函数，写在 `main.c` 中的 `testA`, `testB`, `testC` 能够调用只是因为它们虽然是用户进程但仍然写死在了内核中。**本次实验要求通过系统调用模式打印字符串。**
- 添加两个系统调用 `sys_sem_p` 和 `sys_sem_v`，即信号量的 PV 操作，在此基础上模拟睡眠的理发师问题。
- 共有五个进程，A 进程普通进程，B 进程是理发师，C 进程是顾客，D 进程是顾客，E 进程是顾客。要求有一把理发椅，并支持等待椅子的数目分别为 1、2、3（必须都能够支持，并且可以现场修改，助教检查时在其中随机选择数目），开始时理发师处于沉睡状态。理发师理发消耗两个时间片。
- 普通进程、理发师进程和各个顾客进程需要用不同颜色输出，其中顾客要打印递增的顾客 ID，并打印基本操作比如理发师剪发，顾客得到服务，顾客到来并等待，顾客离开等。

### 注意事项

- 你可能有疑惑要是所有进程都 `sleep` 了时间片给谁了？这确实是个问题。如果四个进程都调用的 `sleep`，在目前《Orange's》的代码上要完美解决可能的改动难度较大。**所以我们假定 A 进程不会调用 `sleep`。**
- 使用 `make` 或类似工具构建整个项目。**其中 Makefile 必须支持 `make run` 命令直接启动程序，不需要其他命令。**
- 本次作业可以直接在《Orange's》的源代码基础上完成，请记录下添加或修改的地方。
- **提交代码、Makefile、说明文档和运行截图。**

### 评分标准

本次实验完成上述要求即可取得全部分数。

## 2. 问题清单

在整个实验的过程中，无论是编程还是查资料，请各位同学注意思考以下问题，助教检查时会从中随机抽取数个题目进行提问，根据现场作答给出分数。*请注意，我们鼓励自己思考和动手实验，如果能够提供自己的思考结果并辅助以相应的实验结果进行说明，在分数评定上会酌情考虑。*

1. 进程是什么？
2. 进程表是什么？
3. 进程栈是什么？

4. 当寄存器的值已经被保存到进程表内，esp 应指向何处来避免破坏进程表的值？
5. tty 是什么？
6. 不同的 tty 为什么输出输出不同的画面在同一个显示器上？
7. 解释 tty 任务执行过程？
8. tty 结构体中大致包含哪些内容？
9. console 结构体中大致包含哪些内容？

如遇到实验相关问题，请在 Moodle 上发帖或向助教发邮件。