

实验四问题清单

1. 进程是什么？

- 进程是一个具有一定独立功能的程序，是关于某个数据集合的一次运行活动
- 进程是操作系统进行资源分配和调度的一个独立单位
- 一个进程包括五个实体部分，分别是：操作系统管理控制运行程序的数据结构、程序状态字信息，运行程序的内存代码、内存数据、通用寄存器信息

2. 进程表是什么？

- 进程表（PCB）是操作系统用于记录和刻画进程状态及环境信息的数据结构，包含标识信息、现场信息、控制信息
- 借助进程表，操作系统可以全面管理进程的物理实体，刻画进程的执行现状，控制进程的执行

3. 进程栈是什么？

进程运行时自身的堆栈。

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？

专门的内核栈区域。

5. tty 是什么？

原来是指电传打字机，现在也泛指计算机的终端（terminal）设备

- 早期的终端（terminal）是一台独立于计算机的机器（teletype，即 TTY）

6. 不同的 tty 为什么输出不同的画面在同一个显示器上？

显示了显存的不同位置。

7. 解释 tty 任务执行的过程？

1. TTY 初始化，开始循环
2. 对于每一个 TTY 先执行 `tty_do_read()`，它调用 `keyboard_read()` 并将读的字符给 `in_progress()` 处理
3. 如果是要输出的字符，会被 `in_progress()` 放入当前接受处理的 TTY 的缓冲区
4. 再执行 `tty_do_write()`，如果缓冲区有数据，就被送入 `out_char()` 显示出来

8. tty 结构体中大概包括哪些内容？

```
1 typedef struct s_tty{
2     u32 in_buf[TTY_IN_BYTES]; /* TTY 输入缓冲区 */
3     u32* p_inbuf_head;        /* 指向缓冲区中下一个空闲位置 */
4     u32* p_inbuf_tail;        /* 指向键盘任务应处理的键值 */
5     int inbuf_count;          /* 缓冲区中已经填充了多少 */
6     struct s_console * p_console;
7 }TTY;
```

9. console 结构体中大概包括哪些内容？

```
1 typedef struct s_console{
```

```
2   unsigned int current_start_addr; /* 当前显示到了什么位置 */
3   unsigned int original_addr;      /* 当前控制台对应显存位置 */
4   unsigned int v_mem_limit;        /* 当前控制台占的显存大小 */
5   unsigned int cursor;             /* 当前光标位置 */
6
7   CURSOR_RECORD cursor_record;     // 存储每一次操作的光标信息，以便退格
8   CHAR_RECORD char_record;         // 存储每一次操作输入的字符，用来撤回
9   unsigned int finalCursorBeforeEsc; // 进入esc前最后一个光标信息，以便退出esc后能返回
10 }CONSOLE;
```

10. 什么是时间片？

分时操作系统分配给每个正在运行的进程微观上的一段 CPU 时间。

11. 结合实验代码解释什么是内核函数？什么是系统调用？

- 系统调用是操作系统提供给用户程序调用的一组“特殊”接口。用户程序可以通过这组“特殊”接口来获得操作系统内核提供的服务。
- 内核函数是指只能在内核模式下调用的例程或子程序，主要是为了内核函数不被用户程序破坏。在操作系统内核中都设置了一组用于实现各种系统功能的子程序，并将它们提供给应用程序调用。