操作系统第三章习题

3.1

使用fork()创建子进程后，子进程拥有自己的内存地址，在子进程中返回值为0，即pid=0，修改value为15，因为子进程中value和父进程的value内存地址不一样，所以父进程的value任为5；而在父进程中，fork()返回的是子进程的进程id>0，直接打印PARENT: value = 5

3.5

C

3.8

短期调度程序(CPU调度程序)——从准备执行的进程中选择进程，并分配CPU。中期调度程序，将进程从内存（或从CPU竞争）中移出，从而降低多道程序程度。之后进程看重新被调入内存，并从中断出继续执行。长期调度程序(作业调度程序)——从进程缓冲池中选择进程加到内存中，以便执行。

他们主要的区别在于执行的频率。短期调度程序必须经常为CPU选择新的进程。长期调度程序并不频繁，因为长期调度程序控制多道程序进程，必须保证多道程序进程的稳定，为此创建创建程序的平均速度必须等于进程离系统的平均速度。所以只有在进程离开系统时才会需要长期调用程序的调度。

3.9

内核会将旧进程状态保存在其PCB中，然后加载经调度后的要执行的新进程的上下文。上下文切换的工作量与硬件支持密切相关，例如有的处理器提供了多个寄存器组，上下文切换只需简单改变寄存器组的指针，但如果活动进程数超过寄存器组数，系统就需要在寄存器和内存之间进行数据复制。

3.12

16个进程

3.13

当子进程创建成功后，调用execlp()加载/bin/ls程序的代码和静态数据，若调用成功，则永远不会执行printf(“line J”)；若调用失败（如不存在/bin/ls这样的可执行程序），才会执行printf(“line J”)

3.14

A: pid=0

B: pid1 = 2603

C: pid = 2603

D: pid = 2600

3.17

LINE X：CHILD: 0 CHILD: -1 CHILD: -4 CHILD: -9 CHILD: -16

LINE Y：PARENT: 0 PARENT: 1 PARENT: 2 PARENT: 3 PARENT: 4

根据 pid，子进程和父进程单独运行。if 块由前者运行，num[5] 中的 0，1，2，3，4 分别乘以 -0，-1，-2，-3，-4，以 0，-1，-4，-9，-16 结尾。else if 块由后者运行，则按顺序输出 num 数组。