

OS HW_2

罗兴攀 PB19051150

1. 创建了15个子进程,加上初始父进程共16个

$$4+6+4+1=15$$

2.

当系统开始执行子进程，即满足`pid==0`条件才会执行`execlp("**")`程序，而只有系统调用`execlp()`函数没有正确执行，即`execlp()`函数内部没有正常执行到`exit()`命令时，`printf("LINE J")`才会被执行。`execlp()`没有正确执行的可能情况为`"/bin/ls"`文件不存在、系统调用中被系统中断等。

3.

A: child: pid=0 #子进程的pid为0

B: child: pid1=2603 #子进程中的pid1是子进程真正的进程编号

C: parent: pid=2603 #父进程的pid是fork()返回的值，是子进程的进程号

D: parent: pid1=2600 #父进程中的pid1是父进程的真实进程编号

4.

LINE X输出: CHILD: 0 CHILD: -1 CHILD: -4 CHILD: -3 CHILD: -16

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

解释: LINE X是子进程输出，子进程会将父进程的num数组复制一份进行计算和输出，不会影响父进程中的数组的值。

5.

原因是`execl("/bin/ls", "/bin/ls", NULL)`没能正确执行。可能是`"/bin/ls"`文件夹缺失或出错、系统调用过程中被系统终端等原因导致

6.

当一个子进程结束运行后，将释放其占据的绝大部分资源（清楚PCB中的大部分数据），但不会完全清空PCB，且这时进程处于“终止态”，“终止态”存在的原因是，父进程可能需要了解子进程的一些信息，例如运行时间、打开的文件等，不能立即清空子进程的PCB（彻底杀死子进程），子进程会返回一个SIGCHLD给父进程，需要由父进程来决定何时彻底杀死子进程，因此需要一个“终止态”来表示子进程等待父进程“杀死”命令的状态。

7.

已经结束运行，但没有彻底消失，而是仅保存基础信息，等待被父进程彻底杀死的进程成为僵尸进程。子进程结束运行后会返回一个SIGCHLD给父进程，父进程对SIGCHLD进行处理，然后将僵尸进程彻底杀死，将该进程的PCB从进程队列中移除并回收资源，此时僵尸进程就消失了。

8.

本地变量、全局变量、动态分配的变量以及程序代码保存在用户空间，而系统调用函数的实现代码、进程控制模块（PCB）则保存在内核空间中。

9.

exec()系统调用与普通函数的最关键区别在于，系统调用会是进程进入到内核态进行执行，拥有对硬件资源的直接访问权力，而普通函数只在用户态执行，用户态访问、使用硬件资源，例如读写文件等其实都是通过系统调用间接访问的。

10.

多线程的优点：

- 响应性更优
- 资源共享更简单，线程间共享所属进程的内存和资源
- 更经济。进程创建所需的内存和资源分配非常昂贵，而线程则经济很多
- 可伸缩性。由于线程可以在多处理器上并行运行，因此多线程对于多处理器结构有很大优势。

堆内存和全局变量是共享的。寄存器值和栈是独立的（不共享的）

11.

a. 创建了5个进程，加上初始父进程总共有6个进程

b. 两个线程被创建。

12.

LINE C: CHILD: value = 5

LINE P: PARENT: value = 0

解释：线程共享同一个进程的数据，但是LINE C和LINE P并非在一个进程中。

13.

- 匿名管道只能用在一对父子进程之间，而命名管道可以用于任意两个进程之间，也可以用在多个进程之间；
- 尽管匿名和命名管道都是半双工通信，但一个匿名管道只能单向通信，而命名管道可以双向通信。（半双工是指虽然可以双向传输，但一个时间点只能一个方向传输。一个匿名管道只能是父到子或子到父的其中一种方向，而一个命名管道可以在不同时间点有不同的通信方向）
- 匿名管道在通信结束后就不存在了，而命名管道不会自动消亡直到程序主动清楚其内存