**实验四：进程间通信设计报告（阐述设计程序的思路）**

**4\_1\_2.c:**

在程序中使用信号需要用到signal.h头文件，同时这里我借鉴了pthread\_t的更好的版本pid\_t(两者都是表示进程号)，不过创建进程使用的就是fork函数了。接着就是用pid记录下进程的pid号，并判断pid是大于，等于还是小于0，对应于不同的情况。pid<0表示进程创建失败；pid等于0表示创建的是子进程，首先输出子进程的pid号，再不断循环执行子进程；pid大于0表示创建的是父进程，睡眠2秒后就会杀死子进程

**4\_2\_2.c:**

使用管道（pipe）首先就要定义一个长度为2的数组int fd[2]，用来记录读或者写的文件描述符；接着创建出BUFFER\_SIZE大小的空间用来保存之后会写入的信息；pipe（fd）用来创建管道；fork用来创建进程。在子进程中，先关闭读端(这样保证正常写)，然后保存写入的msg到管道；在父进程中，先关闭写端(这样保证正常读)，再用read函数从管道中读取数据，于是父进程可以收到子进程发来的消息。

**4\_3\_1.c:**

创建了一个共享存储区，然后让父进程先将数据写入共享存储区内，之后创建两个子进程，每个子进程都向这个共享存储区读出和写入自己的内容，最后父进程将共享存储区中的内容输出。首先要加入需要的头文件<sys/shm.h>，然后在main函数中定义了一个IPC key，而shmid保存共享内存区域的标识符，用来记录shmget函数的返回值。shmget函数创建一个共享存储区，而shmat函数可以将进程连接到共享存储区，sprintf函数用来将各个进程写入的数据到共享存储区。接着调用fork创建两个子进程，并在子进程中写入自己的内容。shmdt用于将当前进程与共享存储区分离；父进程先调用wait函数等待子进程执行完毕，再读出子进程修改后共享存储区的内容。