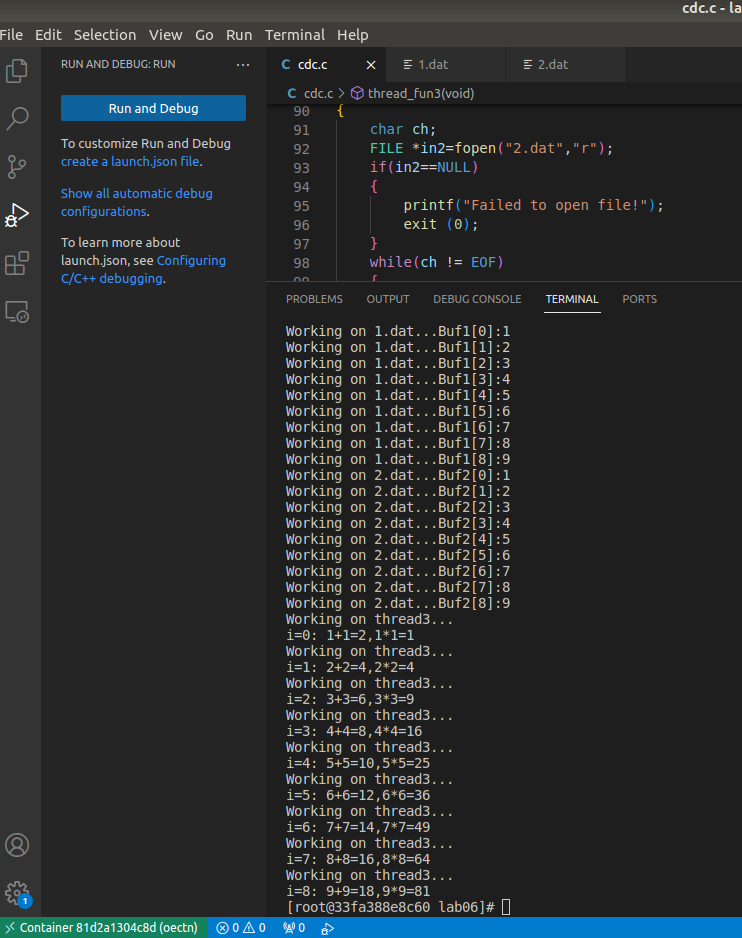
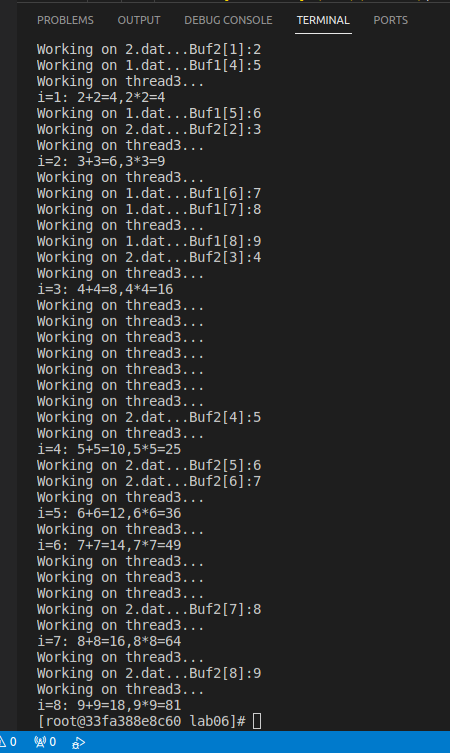
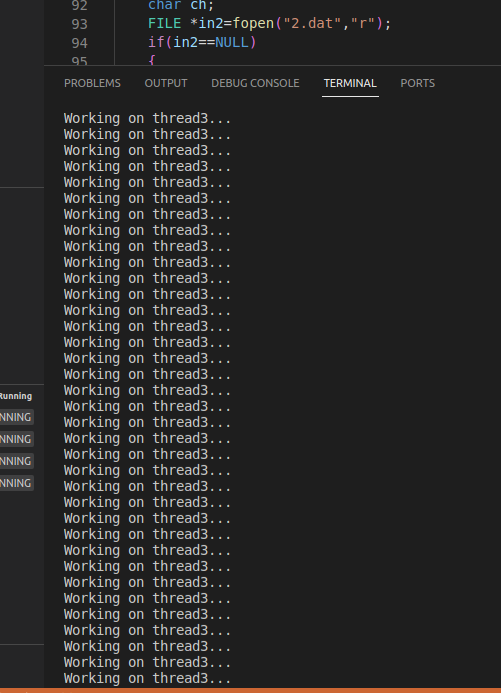
|  |  |  |  |
| --- | --- | --- | --- |
| 课程名称： | 操作系统 | 姓名： | 张祺珲 |
| 实验名称： | Lab6-Semaphore | 学号： | 2020141090128 |
| 任课教师： | 左航 | 专业： | 软件工程 |
| 助教： | 刘俊才 | 班级： | 213110106 |
| 实验成绩 |  | 日期 | 10.15 |

**1.（提交1）完成代码cdc.c，提交运行截图，简单是否可能出现死锁并分析原因。**

1.不添加sleep()



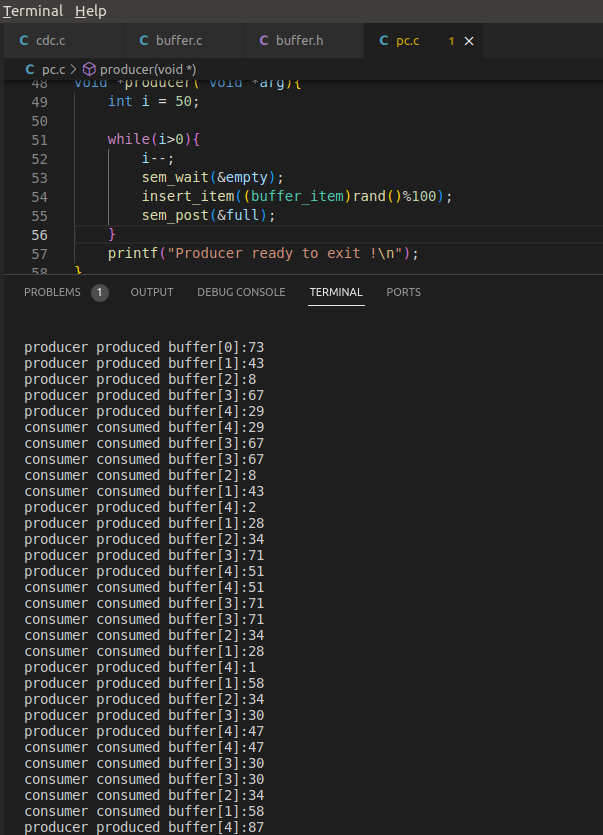
1. 加入不同的sleep之后：



3.有可能出现死锁，比如线程3对线程1进行了加锁操作之后，又相对线程2进行加锁操作，但此时线程2也想进行加锁，线程3和线程2同时加锁，两个线程都没办法抢到锁，而此时线程1又被锁住了，导致三个线程都没法运行，造成了死锁。即由于线程三的占用，Buf1[i]与Buf2[i]的值不能进入互斥区得到输入，线程三也无法结束占用，故死锁

2.**（提交2）提交pc.c文件，运行截图**

修改后的运行截图：



1. 简答题：

**1.请说出练习a,c的相同和不同之处（此处指未移除临界区互斥保护的程序c）。**

1.相同之处：a、c中都形成了临界区，让两个线程不会同时对同一资源进行修改，起到互斥的作用。

2.不同之处：a中使用的是互斥锁来形成临界区，而c中用的是信号量代替了互斥锁来形成临界区。

**2.请说出练习b,c的相同和不同之处（此处指未移除临界区互斥保护的程序b和c）。**

1.相同之处：都用到了信号量，用于指示缓冲区的“空满状态”，让两个线程可以有顺序的访问缓冲区，解决资源数目问题；同时用到二元信号量来建立一个互斥区，让两个线程不会同时访问同一个缓冲区

2.不同之处：b中empty的初始值为5不为1，用信号量管理的资源数目为多个，实现资源的同步；但是在c中缓冲区数目为1，此时相当于起到互斥锁的作用

**3.请解释练习b“分析与总结探索” 移除临界区互斥后为什么会出现不正常的现象**?

因为在b中缓冲区的大小为5，那么可能会出现full和empty同时-1后，同时大于0，那么消费者和生产者就可以同时对临界区进行访问，就有可能存在生产者还没有生产，消费者就已经去消费的情况，所以就会出现-1。

**4.你是否推荐使用二元信号量来代替互斥锁？为什么？**

不推荐，因为互斥锁是设计来专门用于设置临界区的变量，在设计时为了避免出现优先级翻转的情况，加入了优先级继承的机制，从而让CPU的资源能够得到更加合理的使用。但是二元信号量本身并不是为设置临界区设计的，不存在优先级继承的机制，就可能会出现优先级翻转的情况，从而可能导致优先级高的线程得不到CPU的时间，造成CPU的分配不合理的现象。所以不推荐使用二元信号量来代替互斥锁。