# 实验4 信号量

# 实验目的

加强对进程概念的理解。

进一步了解并发执行的实质。

分析进程争用资源的现象，学习解决进程互斥的方法

了解Linux系统中进程通信的基本原理

# 实验内容

使用信号量实现有限缓冲区的生产者和消费者问题

使用信号量实现读进程具有优先权的读者和写者问题

# 实验环境

Ubuntu 22.04.4 LTS

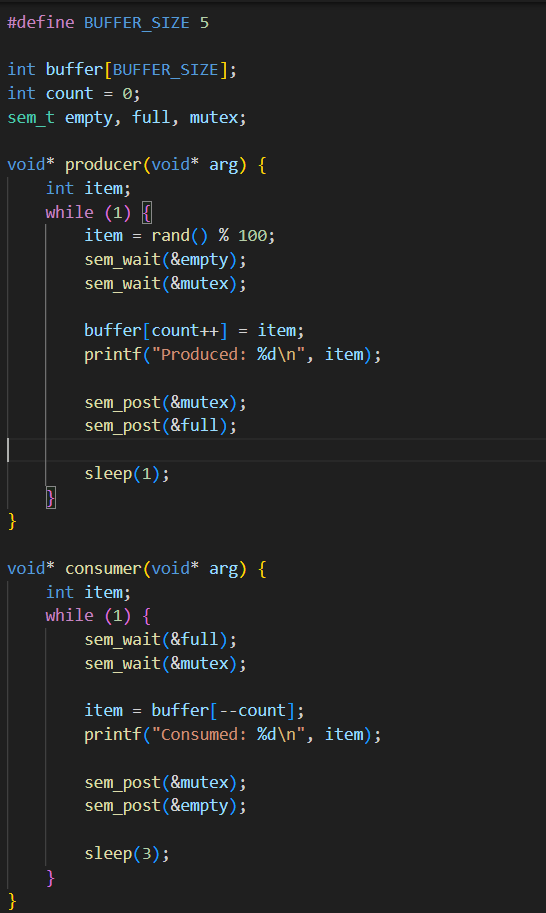
gcc version 11.4.0

# 实验中遇到的主要问题及其解决方式

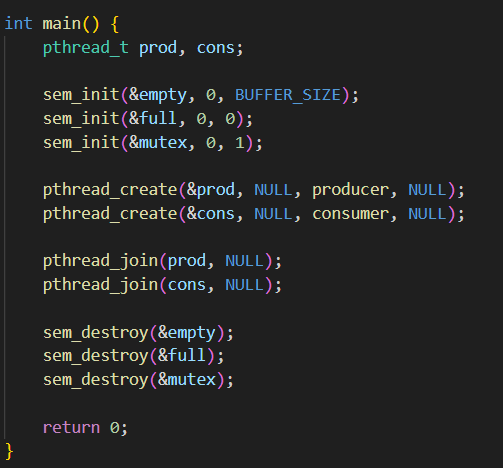
不熟悉C中信号和线程库的函数-查找相关资料熟悉使用

# 源代码

1. 使用信号量实现有限缓冲区的生产者和消费者问题

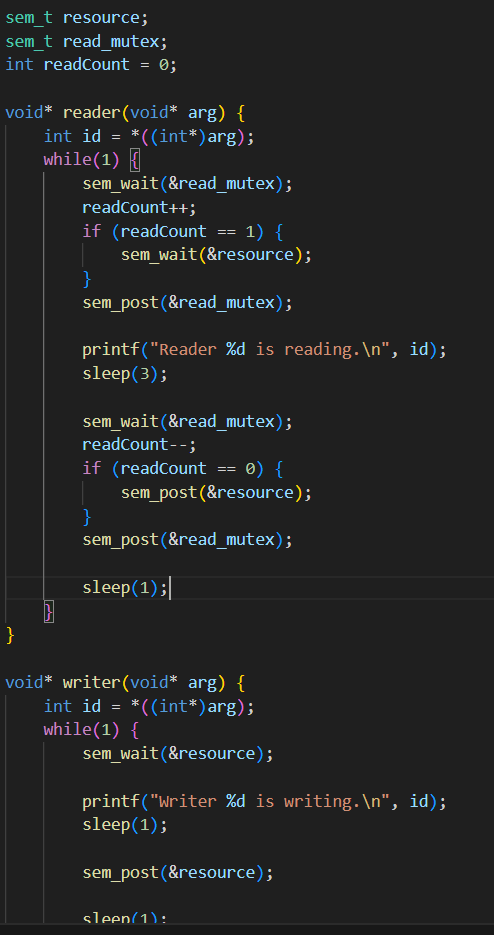


两个线程的task的定义

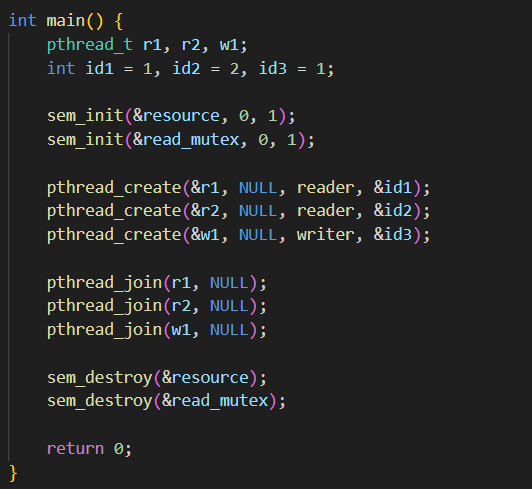


主函数为初始化和销毁自定信号量和创建运行生产者消费者

2. 使用信号量实现读进程具有优先权的读者和写者问题



信号量以及读写进程的任务，没有真正实现读写，但完成了读者优先，这里设置读者占用更长时间



主函数类似，两个读者一个写者

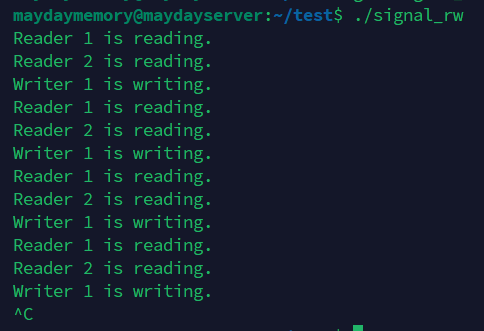
# 实验结果

1. 使用信号量实现有限缓冲区的生产者和消费者问题



手动配置了生产者和消费者速率不同，刚开始生产者运行频次高，达到缓冲区上限后则需要消费者消费使缓冲区有空余才能生产

2. 使用信号量实现读进程具有优先权的读者和写者问题



写者总是在两个读者读完才能访问资源，并且读者占用时间更长，可以明显观察到写者等待的情形

# 实验小结

本次实验通过编写两个程序，分别实现了经典的有限缓冲区的生产者和消费者问题，以及读进程具有优先权的读者和写者问题，重点探索了信号量在进程同步中的应用。使用信号量实现有限缓冲区的生产者和消费者问题 实验展示了通过信号量来协调生产者和消费者对有限缓冲区的访问，确保在任何时刻，缓冲区不会出现溢出或空读的现象。通过模拟生产和消费操作，使生产者和消费者能够在不同步的情况下有效协作，体现了信号量在解决并发问题中的重要性。使用信号量实现读进程具有优先权的读者和写者问题 实验实现了读者优先的同步机制，确保在有读者访问时，写者进程会被暂时阻塞，防止数据的不一致和竞争。读者优先的策略让多个读者可以同时访问共享资源，而写者则在资源完全空闲时独占访问，展现了信号量灵活控制进程访问权限的强大功能。通过这次实验，对信号量在进程同步、资源调度中的应用有了更加深刻的理解。信号量不仅可以有效避免竞争条件和资源冲突，还可以提高系统的并发性能和稳定性。这些技术在实际系统编程中有广泛的应用前景，也为解决复杂的并发问题提供了有力的工具。