# 实验1 进程管理

## 一、实验目的

（1）通过进程的描述与状态转换的模拟实现理解操作系统的进程描述方法和状态转换的原理；

（2）学会进程的创建方法；

（3）通过进程调度的模拟实现分析进程调度算法，理解进程调度的原理；

（4）通过进程切换的模拟实现体会进程切换过程，理解进程切换的原理。

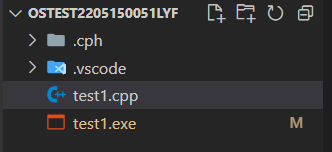
## 二、实验开发环境

系统环境：windows 11

运行环境：vscode

测试环境：vscode

## 三、源程序文件和源码清单



## 四、实验内容

（1）定义PCB结构

包括理论PCB中的基本内容，如内部ID、外部ID、状态、要求服务时间、队列指针等。在实验中只需建立PCB节点，并用它代表一个完整的进程。每创建一个进程时，可动态分配PCB节点，对相应内容赋值，并链接到适当的队列上。

（2）功能实现

1.创建进程

新建一个结构体指针，然后输入该进程的相关数据，再将该指针连接到就绪队列的尾部。

2.时间片到

首先判断就绪队列是否为空，为空则没有进程可以执行。就绪队列的头部节点即为正在执行的进程，该事件发生时，执行进程的时间减去时间片的大小，并将该进程移到就绪队列尾部；如果在时间片时间内进程执行结束，则将该进程结束，并调用下一个节点的进程减去时间片剩余时间。

3.进程阻塞

首先判断就绪队列是否为空，为空则没有进程可以阻塞。若不为空，则将就绪队列头部节点状态改变，并将该节点从就绪队列的头部移到阻塞队列的尾部。

4.唤醒进程

首先判断阻塞队列是否为空，为空则没有进程可以唤醒。若不为空，则将阻塞队列头部节点状态改变，并将该节点从阻塞队列的头部移到就绪队列的尾部。

5.结束进程

根据输入的进程内部id，遍历就绪队列和阻塞队列，若找到该进程，则将该进程删除，若没有，则提示未找到该进程。

6.事件显示

每次操作后，显示正在执行的进程，即就绪队列头部节点，并遍历就绪队列和阻塞队列，打印队列节点，若队列为空，则显示NULL。

7.其他函数

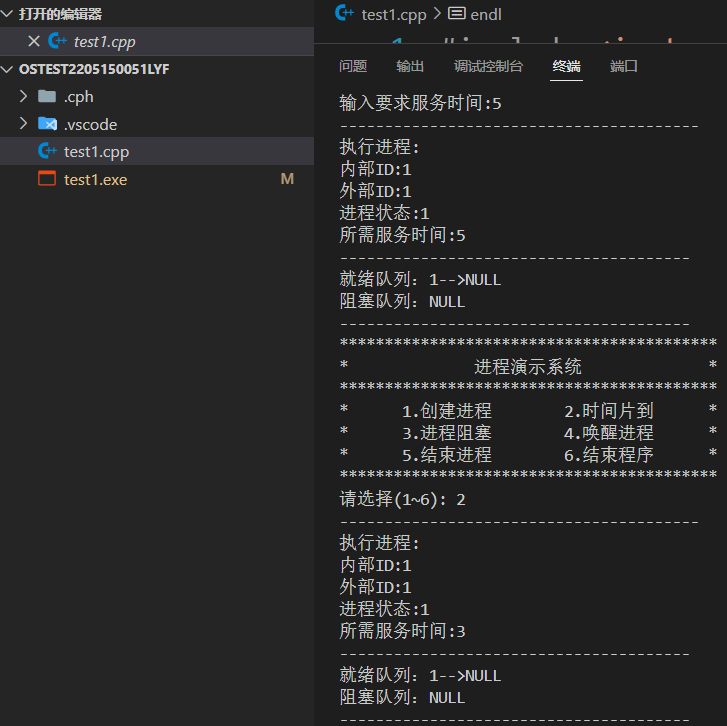
由于实验中的就绪队列和阻塞队列使用链表实现，所以要编写链表的插入，删除和打印函数。

（3）运行结果

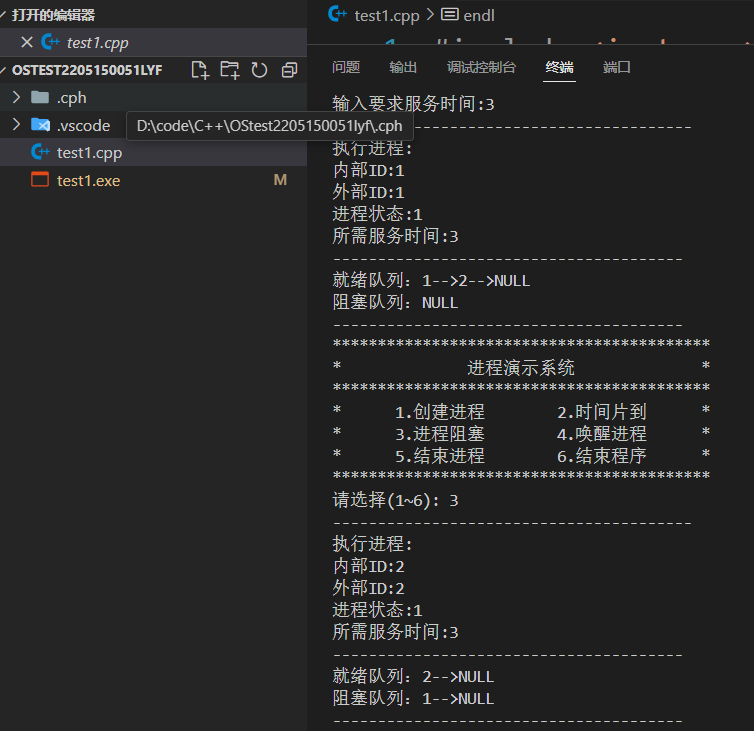
1.创建进程



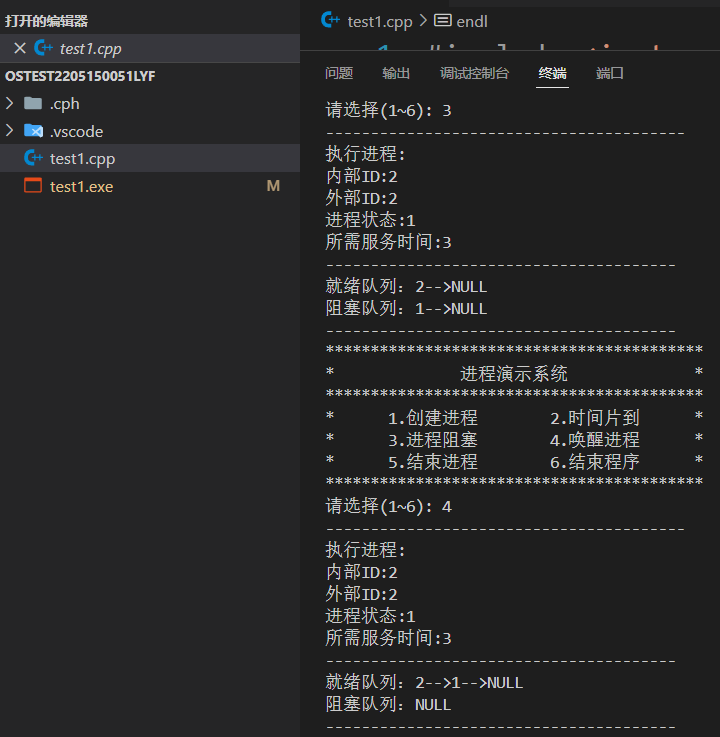
2.时间片到



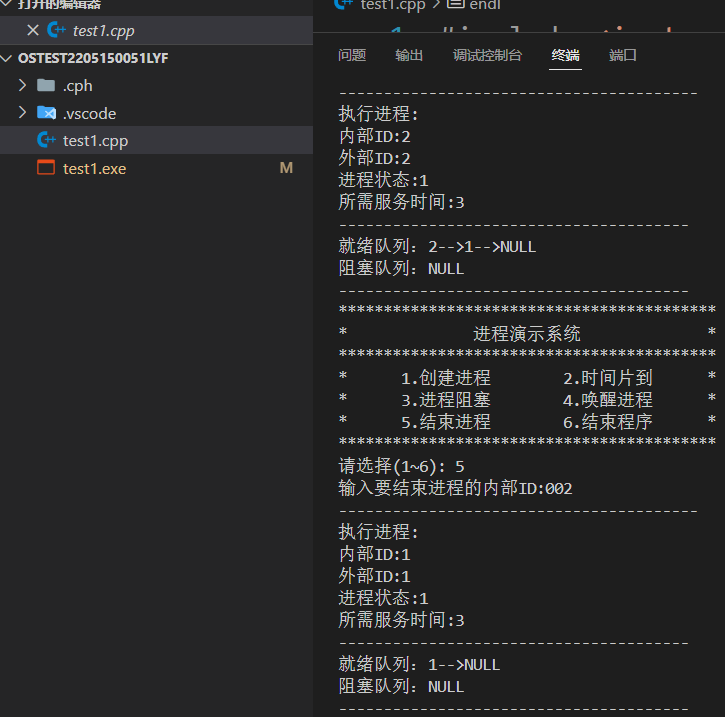
3.进程阻塞



4.唤醒进程



5.结束进程



## 五、总结体会

通过这次实验，我对进程的状态转变和调度算法有了更深入的理解，认识到不同调度策略对系统性能的影响，在实验过程中遇到了一些实际问题，例如死锁和资源竞争，通过查阅资料和团队讨论，逐步找到了合理的解决方案。