# 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、实验内容

1、基本要求：

利用简单的结构和控制方法模拟进程结构、进程状态和进程控制。要求必须完成以下几个方面的内容：

（1）能够模拟进程的创建与撤销过程；

（2）可对进程的状态进行全面的控制；

（3）能够模拟时间片轮转调度算法进行进程调度的过程；

（4）按先进先出方式管理就绪和阻塞队列，能够按队列形式输出进程状态。

2、实验提示：

（1）用PCB表示整个进程实体，利用随机数方法或键盘控制方法模拟进程执行中产生的事件，或者利用基于图形界面的鼠标或者键盘操作控制进程管理内容。

（2）定义PCB：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。由于很难实现真正的进程创建功能，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。

可参考如下数据结构：

struct PCB{

int in\_id

char out\_id[10];

int state;

int run\_time;

struct PCB\*next;

};

struct PCB \*ready\_head, \*blocked\_head, \*ready\_tail, \*blocked\_tail, \*running;

其中就绪队列和阻塞队列是带头指针和尾指针的带头结点的单链表，running指向当前执行进程的PCB。

创建进程时：

struct PCB \*p=(struct PCB \* )malloc(sizeof(struct PCB));

并从键盘输入此进程PCB的基本信息，将此进程PCB添加到就绪队列末尾：

（3）定义进程状态转换方式：真实的进程状态转换是由进程内部操作或操作系统的控制引起的。由于模拟程序中无法实现这些功能，我们可以采用**随机数方法**或**键盘控制方法**模拟，并实现对应的控制程序。

**随机数方法**指产生1－5的随机数，分别代表(1)创建进程、(2)时间片到、(3)进程阻塞、(4)唤醒进程、(5)结束进程等事件；**键盘模拟方法**指定义5个选项菜单代表以上五种事件。其中使用**键盘模拟方法**更便于观察运行结果。

注：“(5)结束进程”是指外界干预强制结束进程，该进程可能处于任何状态。

（4）根据随机数或键盘操作处理就绪队列、阻塞队列和当前执行进程的状态。**每次事件处理后应显示出当前系统中的执行进程是哪一个，就绪队列和阻塞队列分别包含哪些进程**。

**三、实验报告撰写和提交要求**

实验报告撰写和提交要求：

（1）实验报告内容，须涵盖开发环境、运行环境、测试环境、源程序文件及源码清单、实验步骤、技术难点及解决方案、关键数据结构和算法流程、编译运行测试过程及结果截图、疑难解惑及经验教训、结论与体会等；

（2）在实验报告内容（如运行结果截图等适当位置）中应有机融入个人学号、姓名、计算机系统信息等凸显个人标记特征的信息；

（3）独立完成实验任务，整理出实验报告电子档(以“学号+姓名+实验序号”命名)，实验报告电子文档提交格式可为 Word 文档、WPS 文档或 PDF 文档，以附件形式提交到学习通，并发给学习委员。

（4）实验报告提交截止时间为实验结束后一周内。