

# 2021-2022(第二学期) 操作系统课程设计

## 一、课程性质

操作系统课程设计是计算机科学与技术专业理解和巩固操作系统基本理论、原理和方法的专业实践环节，通过操作系统课程设计，综合应用学生所学知识，建立完整的计算机系统概念，理解和巩固操作系统基本理论、原理和方法，掌握操作系统基本理论与管理方式。培养学生严谨的科学态度，严肃认真的工作作风，和团队协作精神。

通过本实践环节的锻炼，使学生能够综合应用学生所学知识，建立系统和完整的计算机系统概念，理解和巩固操作系统基本理论、原理和方法，掌握操作系统基本理论与管理方式。在算法基础上，解决实际的管理功能的问题，提高学生实际应用、编程的能力。

## 二、课设题目

### 1. 题目名称：进程间的互斥与同步

**内容要求：**编写算法，实现进程间对临界资源的互斥访问以及进程间的同步关系。具体实现要求如下：（不局限下述情况，在工作量不缩减的情况下，可行设计实现过程。）

- （1）要求进程互斥使用文本文件；
- （2）假定文本文件 txt1 最大可写入 30 个字符；
- （3）写满后复制进程将文本文件的内容复制到另一个文本文件 txt2 中（无长度限制）；
- （4）复制进程复制完毕写入进程可再重新写入，重复执行（3）、（4），直到得到停止命令；
- （5）实现进程间的同步和互斥。

### 2. 题目名称：多进程共享临界资源

**内容要求：**编写算法，模拟实现多进程间对临界资源的共享。具体实现要求如下：（不局限下述情况，在工作量不缩减的情况下，可行设计实现过程。）

- （1）要求至少有三个进程；
- （2）一个进程作为原语的管理进程，接受其他进程的临界区进入请求。如果允许进入，则设置相应变量，然后返回；如果不允许，则进入循环等待，直到允许为止；

(3) 其他进程模拟需要进入临界区的用户进程。当需要进入临界区时向管理进程提出申请，申请返回表示进入了临界区；当退出临界区时（当时间结束时），向管理进程提出推出申请；申请返回表示已退出临界区。

(4) 对临界区的访问应遵循空闲让进、忙则等待、有限等待、让权等待；

(5) 输出进程进入临界区到退出临界区的过程，进入临界区时显示“进程 X 请求进入临界区...”，在临界区等待一段随机时间时显示“进程 X 正在临界区...”，提出退出时显示“进程 X 退出临界区...”。

### 3. 题目名称：进程调度

**内容要求：**编写算法，模拟实现进程调度算法，包括非抢占短进程优先算法、可抢占优先权调度算法及多级反馈队列调度算法。具体实现要求如下：（不局限下述情况，在工作量不缩减的情况下，可行设计实现过程。）

(1) 调度所需的进程参数（即 PCB，包括进程名、到达时间、要求服务时间、优先级等）由输入产生（可手工输入或随机产生）；

(2) 进行算法评价，计算平均周转时间；

(3) 输出调度结果，输出算法评价指标。

### 4. 题目名称：动态资源分配

**内容要求：**编写算法，模拟实现可避免死锁的动态资源分配，即通过对系统资源预分配后检查系统状态（银行家算法）以避免死锁的产生。具体实现要求如下：（不局限下述情况，在工作量不缩减的情况下，可行设计实现过程。）

(1) 资源总类与数目可在界面进行设置，在资源分配过程中可以随时增加进程及对资源的需求；

(2) 在资源分配过程中要进行系统安全状态检查，通过检测则对该进程进行资源分配；

(3) 当进程满足所有资源分配后自行释放所有资源；

(4) 输出资源分配过程。

### 5. 题目名称：存储器管理

**内容要求：**编写算法，模拟实现动态分区存储管理。具体实现要求如下：（不局限下述情况，在工作量不缩减的情况下，可行设计实现过程。）

(1) 算法所需的参数由输入产生：可由手工输入也可基于随机数产生；

(2) 采用空闲链结构描述空闲分区和已分配分区情况；

(3) 设计主页面以灵活选择分区分配算法，要求以下算法都要实现：首次适应、循环首次适应、最佳适应、最坏适应；

(4) 内存回收（多种情况）要考虑全部情况。

## 6. 题目名称：虚拟存储器管理

**内容要求：**编写算法，模拟实现请求分页存储管理，实现地址变换及页面置换。具体实现要求如下：（不局限下述情况，在工作量不缩减的情况下，可行设计实现过程。）

（1）算法所需的参数由输入产生：可由手工输入也可基于随机数产生；产生的数据作为逻辑地址；

（2）假定一页大小为 1 K，内存是 10K；

（3）如果访问页面在内存计算物理地址（考虑地址的合法性）；若不在则调入或置换（采用改进型时钟置换算法）；假定初始时页面均不在内存。

（4）输出内存驻留的页面集合，展示页面访问过程。

## 7. 题目名称：SPOOLING 输出模拟系统

**内容要求：**编写算法，实现对 SPOOLING 输出模拟系统。具体实现要求如下：（不局限下述情况，在工作量不缩减的情况下，可行设计实现过程。）

（1）包括一个 SPOOLING 输出进程和一个 SPOOLING 输出请求服务程序以及两个请求输出的进程。

（2）请求输出进程每次调用请求服务程序输出一行信息，由输出请求服务程序将该信息项送如输出井。待输出一个结束标志时，表示一个输出文件输出完成，在输出井中形成一个输出信息块，并构成一个输出请求块，SPOOLING 输出进程工作时，根据某输出请求块将输出井中相应信息块实际输出到打印机或 CRT，SPOOLING 进程与请求输出进程可并发运行。

（3）进程调度采用随机调度法，这与进程的输出的随机性相一致。两个请求输出进程的调度概率各为 45%，SPOOLING 输出进程为 10%，这由产生随机数来模拟。

（4）为进程设置三种基本运行状态：可执行、等待和结束。等待状态又可分成等待状态 1 和等待状态 2。状态变换的条件如下：

①进程执行完毕后置成“结束态”。

②服务程序在输出信息时，如发现输出井已满，将调用进程置成“等待状态 1”。

③SPOOLING 进程在输出井空时进入“等待状态 2”。

④SPOOLING 进程输出一个信息块后，应释放该信息块所占用的输出井空间，并将正在等待输出的进程置为“可执行态”。

⑤ 服务程序在输出信息到输出井并形成信息块后，将 SPOOLING 进程置成“可执行态”。

## 8: 题目名称：磁盘空间的管理

**内容要求：**编写算法，模拟实现磁盘空间管理，为文件分配盘块。具体实现要求如下：（不局限下述情况，在工作量不缩减的情况下，可行设计实现过程。）

- （1）采用位示图管理外存空间；
- （2）建立一个磁盘控制块用于描述整个磁盘空间的使用情况（如磁盘的空闲块数、空闲字节数）；
- （3）设外存空间大小为 512 字节，用位示图表示空间的使用情况；
- （4）文件可采用顺序或索引的方式表示；
- （5）当对文件进行操作时（如 create,write,delete 等），完成磁盘的分配和回收工作，能够实现文件的动态增长；
- （6）输出要显示文件的内容和磁盘空间的使用情况。

#### **9. 题目名称：交通信号灯模拟**

**内容要求：**编写算法，模拟实现交通信号灯的控制。假设一个十字路口，共有四组红绿灯，每个路口的车辆都遵循“红灯停、绿灯行”的原则，将每一台汽车都作为一个进程，请设计算法实现合理的“十字路口交通管理”情况。具体实现要求如下：（不局限下述情况，在工作量不缩减的情况下，可行设计实现过程。）

- （1）车辆数目：每个路口的车辆数目不超过 5 辆，程序能够提供参数来设定初始的车辆数目；
- （2）交通灯设定：初始情况一个路口为红灯、一个路口为绿灯；红灯变为绿灯的间隔为 3 秒，必须保持十字路口交通灯的设定正确；
- （3）车辆通行设定：对一个路口而言，只有当一辆车通过路口（越过对面路口的交通灯后）后，其后续车辆才能继续通过交通灯，车辆通过路口的时间如下：（不局限下述情况，在工作量不缩减的情况下，可行设计实现过程。）可以固定、可以自行计算，但是不能为 3 秒（即必须体现交通灯对通行的管制）。

#### **10. 题目名称：虚拟存储的请求分页管理模拟**

**内容要求：**请求页式管理是一种常用的虚拟存储管理技术。本设计通过请求页式存储管理中页面置换算法模拟设计，了解虚拟存储技术的特点，掌握请求页式管理的页面置换算法。

通过随机数产生一个指令序列，共 320 条指令。指令的地址按下述原则生成：

- ① 50% 的指令是顺序执行的；
- ② 25% 的指令是均匀分布在前地址部分；
- ③ 25% 的指令是均匀分布在后地址部分。

具体的实施方法是：

- ① 在 [0,319] 的指令地址之间随机选取一起点 m；
- ② 顺序执行一条指令；

- ③ 在前地址 $[0, m+1]$ 中随机选取一条指令并执行，该指令的地址为  $m'$ ；
- ④ 顺序执行一条指令，其地址为  $m'+1$ ；
- ⑤ 在后地址  $[m'+2, 319]$  中随机选取一条指令并执行；
- ⑥ 重复上述步骤② ~ ⑤，直到执行 320 次指令。

将指令序列变换成为页地址流设：

- ① 页面大小为 1K；
- ② 用户内存容量为 4 页到 32 页；
- ③ 用户虚存容量为 32K。

在用户虚存中，按每 K 存放 10 条指令排列虚存地址，即 320 条指令在虚存中的存放方式为：

第 0 条 ~ 第 9 条指令为第 0 页（对应虚存地址为  $[0, 9]$ ）；

第 10 条 ~ 第 19 条指令为第 1 页（对应虚存地址为  $[10, 19]$ ）；

⋮

第 310 条 ~ 第 319 条指令为第 31 页（对应虚存地址为  $[310, 319]$ ）。

按以上方式，用户指令可组成 32 页。

计算并输出下述各种算法在不同内存容量下的命中率。

先进先出的算法 (FIFO)；

最近最少使用算法 (LRR)；

最少访问页面算法 (LFR)；

最近最不经常使用算法 (NUR)。

提示

1. 命中率 =  $1 - \frac{\text{页面失效次数}}{\text{页地址流长度}}$

在本实验中，页地址流长度为 320，页面失效次数为每次访问相应指令时，该指令所对应的页不在内存的次数。

2. 随机数产生办法

关于随机数产生办法，Windows 系统提供函数 `srand()` 和 `rand()`，进行初始化和产生随机数。例如：

```
#include <time.h>

srand( (unsigned)time( NULL ) );

语句可初始化一个随机数；

a[0]=10 * rand() / 32767 * 319 + 1;
a[1]= 10 * rand() / 32767 * a[0];
⋮
```

语句可用来产生  $a[0]$  与  $a[1]$  中的随机数。

### 11. 题目名称：多线程并行访问控制

**内容要求：**编写算法，模拟实现多线程并行访问控制。具体实现要求如下：（不局限下述情况，在工作量不缩减的情况下，可行设计实现过程。）

- （1）创建3个服务器线程。
- （2）创建1个服务器并行管理线程。
- （3）创建10个客户线程，随机向服务管理线程提出访问请求，如果没有得到响应，随机选择一段时间重新发出请求。
- （4）显示服务线程与客户线程的对应关系。

### 12. 题目名称：动态分区分配方式模拟

**内容要求：**编写算法，模拟实现动态分区分配。具体实现要求：：（不局限下述情况，在工作量不缩减的情况下，可行设计实现过程。）

（1）分别采用首次适应算法和最佳适应算法实现动态分区分配过程 `alloc( )` 和回收过程 `free( )`。其中，空闲分区通过空闲分区链表来管理，在进行内存分配时，系统优先使用空闲区低端的空间。

（2）假设初始状态如下，可用的内存空间为 640KB，并有下列的请求序列；

作业 1 申请 130KB

作业 2 申请 60KB

作业 3 申请 100KB

作业 2 释放 60KB

作业 4 申请 200 KB

作业 3 释放 100 KB

作业 1 释放 130 KB

作业 5 申请 140 KB

作业 6 申请 60 KB

作业 7 申请 50KB

作业 6 释放 60 KB

（3）请在完成内存块的分配和回收的同时，显示内存块分配和回收后空闲内存分区链的情况。

（4）比较采用首次适应算法和最佳适应算法，对内存的分配和回收速度会造成什么不同的影响？

（5）如何解决因碎片而造成内存分配速度降低的问题？

### 13 题目名称：进程调度模拟算法

**内容要求：**编写算法，模拟实现进程调度。具体实现要求：（不局限下述情况，在工作量不缩减的情况下，可行设计实现过程。）

（1）实现对 N 个进程采用动态优先权优先算法的进程调度。

（2）每个用来标识进程的进程控制块 PCB 用结构来描述，包括以下字段：

- 进程标识数 ID；
- 进程优先数 PRIORITY，并规定优先数越大的进程，其优先权越高；
- 进程已占用的 CPU 时间 CPUTIME；

- 进程还需占用的 CPU 时间 ALLTIME。当进程运行完毕时，ALLTIME 变为 0；
- 进程的阻塞时间 STARTBLOCK，表示当进程再运行 STARTBLOCK 个时间片后，进程将进入阻塞状态；
- 进程被阻塞的时间 BLOCKTIME，表示已阻塞的进程再等待 BLOCKTIME 个时间片后，进程将转换成就绪状态；
- 进程状态 STATE；
- 队列指针 NEXT，用来将 PCB 排成队列。

(3) 优先数改变的原则：

- 进程在就绪队列中呆一个时间片，优先数增加 1；
- 进程每运行一个时间片，优先数减 3。

(4) 假设在调度前，系统中有 5 个进程，它们的初始状态如下：

ID	0	1	2	3	4
PRIORITY	9	38	30	29	0
CPUTIME	0	0	0	0	0
ALLTIME	3	3	6	3	4
STARTBLOCK	2	-1	-1	-1	-1
BLOCKTIME	3	0	0	0	0
STATE	READY	READY	READY	READY	READY

(5) 为了清楚地观察进程的调度过程，程序应将每个时间片内的进程的情况显示出来，参照的具体格式如下：

```

RUNNING PROG: i
READY_QUEUE:->id1->id2
BLOCK_QUEUE:->id3->id4

=====

ID01234
PRIORITY0P0P1P2P3P4
CPUTIMEC0C1C2C3C4
ALLTIMEA0A1A2A3A4
STARTBLOCKT0T1T2T3T4BLOCKTIMEB0B1B2B3B4
STATES0S1S2S3S4

```

#### 14. 题目名称：P、V操作及进程同步实现

**内容要求：**编写算法，用语言编程实现P、V原语并用P、V原语描述如下生产者-消费者问题：有一个理发师，一把理发椅和n把提供给等候理发的顾客座的椅子。如果没有顾客，则理发师便在理发椅子上睡觉；当第一个顾客到来时，必须唤醒该理发师进行理发；如果理发师正在理发时又有顾客到来，则如果有空椅子可坐，他就坐下来等待，如果没有空椅子，他就离开理发店。为理发师和顾客各编一段程序描述他们的行为，要求不能带有竞争条件，试用P、V操作实现。具体实现要求：

- (1) 定义信号量并将P、V操作定义为带参数
- (2) 以输出字符串的形式表示理发师和顾客的行为。
- (3) 设计适当的数据结构和函数描述顾客等待队列和“唤醒”理发师理发过程，



以及没有顾客时的“阻塞”理发师过程。

(4) 编程时需考虑理发师和顾客对应的程序是并发操作的。提示：可利用随机函数模拟并发操作。

(5) 理发师和顾客两个进程各自调用一个函数模拟生产及消费的操作。消费者进程开始时首先测试生产者是否存在，若不存在，则循环测试直到生产者出现为止。消费者如果是第一次执行即转为睡眠状态，则直到生产者完成产品后再唤醒消费者，然后两者协调地工作下去。

(6) 假设题目中由2名理发师和2把理发椅，上述程序应做哪些修改？（可选作）

## 10. 题目名称：进程间通信

**内容要求：**编写算法，模拟实现进程间通信。具体实现要求：

(1) 共享存储区的创建，链接和断开

(2) 消息的创建，发送和接收

(3) 编写程序1，实现利用共享存储区进行进程通信。使用系统调用shmget()，shmat()，shmdt()及shmctl()编制一长度为1k的消息发送和接收程序。

(4) 编写程序2，实现利用消息队列进行进程通信。使用系统调用shmget()，shmat()，shmdt()及shmctl()编制一长度为1k的消息发送和接收程序。

<程序设计提示>（不局限下述情况，在工作量不缩减的情况下，可行设计实现过程。）

(1) 为了便于操作和观察结果，用一个程序作为“引子”，先后fork()两个子进程，SERVER和CLIENT，进行通信。

(2) SERVER端建立一个Key为75的共享区，并将第一个字节置为-1，作为数据空的标志，等待其他进程发来的消息。当该字节的值发生变化时，表示收到了消息，进行处理。然后再次把它的值设为-1。如果遇到的值为0，则视为结束信号，取消该队列，并退出SERVER。SERVER每接收到一个消息后显示一句“(server) received”。

(3) CLIENT端使用Key为75的共享区，当共享取得第一个字节为-1时，SERVER端空闲，可发送请求。CLIENT随即填入9到0。期间等待SERVER端再次空闲。进行完这些操作后，CLIENT退出。CLIENT每发送一条信息后显示一句“(client) sent”。

(4) 父进程在SERVER和CLIENT均退出后结束。

## 三、课程设计的基本要求

(1) 围绕着所选题目开展相关文献查询，并基于相关文献资料及课程基础理论知识，对题目进行问题分析，形成设计或研究方案；在此基础上，独立完成相关算法的详细设计及实现。



(2) 以完整的作业包的形式提交**相关文献资料、原始代码、设计文档和可运行程序**（注：存储在文件夹中，**文件夹命名方式“学号+姓名+联系电话”**）。课程设计报告字数**不少于2000字**，主要内容有：

- ① 概述：设计主要完成的任务和解决的主要问题，并将文献中涉及相关研究工作进行归纳总结；2分
- ② 设计的基本概念和原理；2分
- ③ 总体设计：实现的方法和主要技术路线。；4分
- ④ 详细设计：使用主要函数、方法等（要有画出主要模块的程序流程图）；4分
- ⑤ 软件测试：表明采用的测试方法及测试用例；2分
- ⑥ 简要的使用说明；（可选）2分
- ⑦ 总结：设计、实现过程中遇到的问题、经验、教训和感受。2分

四、选题及成绩评定

- 1. 题目选题：每人可从题号1-12中，任选一题，**但每个题目最多3个人。（以选题申报顺序为准。）**
- 2. 考核比例与标准：

答辩、程序演示和讲解	设计报告
<b>50%</b>	<b>50%</b>

- (1) 程序演示和讲解考核标准  
工作态度；系统设计的合理性、科学性、创新性；工具使用的熟练程度。
- (2) 设计报告考核标准  
论述充分，内容齐全，文字通顺，条理清楚，图表规范。
- (3) 答辩考核标准  
回答问题的正确性，叙述是否流畅，思路清晰程度。

3. 成绩考核具体内容与课程目标评价方式表

考核方式	评价方式	课程目标 1	课程目标 2
答辩、演示和讲解 (50 分)	开发工具使用 (10 分)	0.1	0.1
	是否满足题目的要求 (10 分)	0.3	
	系统设计的合理性等 (10 分)		0.3
	独立完成设计 (10 分)		0.2
	答辩表现 (10 分)		0.2
设计报告 (50 分)	内容齐全 (20 分)	0.2	

	论述充分 (10 分)	0.2	0.2
	图表规范, 条理清楚 (20 分)	0.2	

详细评分细则:

考核方式	评价方式		课程目标 1	课程目标 2
答辩、演示和讲解 (50 分)	开发工具使用 (10 分)	熟练使用 (10 分)	0.1	0.1
		基本熟练 (8 分)		
		不熟练 (6 分)		
	是否满足题目的要求 (10 分)	满足要求 (10 分)	0.3	
		基本满足要求 (8 分)		
		不满足要求 (6 分)		
	系统设计的合理性 (10 分)	能够实现功能, 操作便捷 (10 分)		0.3
		能够实现功能, 操作繁琐 (8 分)		
		能够实现部分功能 (6 分)		
		无法实现要求功能 (3 分)		
	独立完成设计 (10 分)	代码规范, 包含注释 (10 分)		0.2
		代码部分不规范, 注释不完整 (8 分)		
		代码不规范, 无注释 (6 分)		
	答辩表现 (10 分)	表述清晰 (5-4-3 分)		0.2
		正确回答问题 (5-4-3 分)		
设计报告 (50 分)	内容齐全 (20 分)	报告书: ① 概述 2 分 ② 设计的基本概念和原理 2 分 ③ 总体设计 3 分 ④ 详细设计 3 分 ⑤ 软件测试 2 分 ⑥ 简要的使用说明 2 分 ⑦ 总结 2 分 文献: 是否详实 2 分 共计 20 分, 缺项扣除相应分数	0.2	
	论述充分 (10 分)	论述内容详实 (10 分)	0.2	0.2
		论述内容基本详实 (8 分)		
		论述内容简略 (6 分)		
	图表规范, 条理清楚 (20 分)	图标规范 (10-8-6 分)	0.2	
		条理清楚 (10-8-6 分)		