

2023 年《 计算机操作系统 》
课程设计实践指导手册
(面向人工智能、数据科学专业)
第 V1 版

南京农业大学人工智能学院
姜海燕

jianghy@njau.edu.cn

QQ 群号：556067749

2023 年 8 月 21 日

目 录

1 课程设计目的与选题	1
名称：仿真实现作业管理与连续动态内存管理	1
对于输入输出类型的作业指令，仿真实现通过内存缓冲区输入输出 ‘’	1
2.基础设计	1
2.1 裸机硬件仿真设计	1
(1) CPU 与寄存器的仿真设计	2
(2) 系统时钟中断仿真	2
(3) 作业请求的中断仿真	2
(4) 进程调度的中断仿真	2
(5) 输入/输出中断事件仿真	3
(6) 用户区内存	3
(7) 内存缓冲区	3
2.2 并发作业请求文件 (jobs=input.txt) 的设计	3
2.3 用户程序指令文件的设计	4
2.4 作业运行及调度详细记录的文件保存	5
2.6 进程控制块 PCB 设计	7
2.7 系统 PCB 表的设计	8
2.8 仿真实现进程控制原语	8
2.9 其他	8
3 2023-2024 年第一学期课程设计时间节点安排	9
3.1 第 1 周 (8 月 28 日-9 月 1 日)，课程设计阅读指导手册，组建讨论组并开展活动； 参考教材规划自己的课设计划，	9
3.2 第 2 周 (9 月 5 日)，周二晚上 6:00-7:00，课程设计线下辅导讲解 (地点另行通知)	9
3.4 第 4 周 (9 月 20 日，周三晚上 7:00-8:00)，课设线上答疑，其他时间修改课设程序	9
3.5 第 5 周 (9 月 27 日，周三晚上 7:00-8:00)，课设线上答疑，其他时间修测试课设程序	9
3.6 第 7 周 (10 月 9 日-13 日，全周停课课设, 地点：浦口校区实验室 (地点另行通知)	9
3.7 正式提交全部材料时间：	9
10 月 12 日 (周四) 提交自己的课设完整材料；	9
10 月 13 日 (周五) 提交互测完整材料	9
3.8 停课 5 天时间安排：	9
4 正式材料提交要求	9
5 课程设计评价标准说明	12
5.1 成绩百分比比例	12
6 课程设计实践报告内容与要求	12
7 课程设计课程联系方式	13

1 课程设计目的与选题

1.1 实验目的

以计算机操作系统原理为指导，利用面向对象程序设计技术仿真 OS 内核的进程管理、作业管理、连续内存管理等 API 函数功能，可视化显示操作系统的工作过程，完成操作系统的仿真程序设计、开发、测试，录屏讲解，同时需要撰写实践报告、自测报告，录制代码设计实现讲解视频、自测过程讲解视频文件。

注意：本课程设计必须按照基础设计要求完成，学有余力同学可以扩展设计，可视化界面需自行设计。

2.2 选题

名称：仿真实现作业管理与连续动态内存管理

◆及格要求（成绩等级：D）

利用 3 个线程（时钟中断、）实现时钟中断、作业并发、进程调度以及输入输出；

多任务作业并发环境下，作业调度算法、进程原语、时间片轮转的进程调度算法实现；
不考虑内存分配，假设内存足够大，创建新进程时不受内存大小限制；

使用 input1 文件夹中的作业为输入；

◆中等要求（成绩等级：C）

完成及格要求前题下，完成以下功能：

实现 MMU 地址变换，按照基础设计实现内存动态分配管理；

将连续动态空间分配过程、作业调度、进程调度过程通过可视化方式呈现；

使用 input1 文件夹中的作业文件为输入；

◆良好要求（成绩等级：B 系列，包括 B、B+）

完成中等要求前题下，完成以下功能：

需要考虑进程调度中阻塞与唤醒操作；

采用多级反馈队列进程调度算法实现进程调度；

接收界面上实时作业生成并参与调度的功能；

使用 input2 文件夹中的作业文件为输入；

仿真过程可视化；

◆优秀要求（成绩等级：A 系列，包括 A、A+）

完成良好要求前题下，完成以下功能：

对于输入输出类型的作业指令，仿真实现通过内存缓冲区输入输出 ‘ ’

扩展 input2 文件夹中作业指令文件的设计，使其能够仿真实现伙伴系统

设计指令文件指令，通过死锁检测算法仿真出进程死锁现象，将实现过程通过可视化方式呈现。

2. 基础设计

2.1 裸机硬件仿真设计

包括 CPU、内存、系统时钟中断、用户内存、内存缓冲区、输入设备中断、输出设备中断、地址变换机构（MMU）等；

(1) CPU 与寄存器的仿真设计

◆CPU 可抽象为一个类，名称：CPU。

◆关键寄存器可抽象为类的属性，至少包括：程序计数器（PC）、指令寄存器（IR）、状态寄存器（PSW）等，寄存器内容的表示方式自行设计，需要在实践报告中说明。

需要实现 CPU 现场保护、现场恢复操作，封装为 CPU 类的方法，供进程切换、CPU 模式切换方法调用。

方法函数的名称须统一，入口出口参数自行定义：

◆CPU-PRO（）：CPU 现场保护函数；

◆CPU-REC（）：CPU 现场恢复函数；

(2) 系统时钟中断仿真

设计自己的计算机系统时钟，其他中断使用该时钟统一时钟计时；

◆设计一个时钟线程类，名称：Clock_thread

◆该类中设计一个共享属性变量，名称：COUNTTIME，整型，单位：秒（s）

◆时钟计时函数，名称统一，入口出口参数自行定义：

◆TIME-COUNT（）：通过该函数对 COUNTTIME 变量计时操作。此变量为临界资源，需要互斥访问；因此，操作该变量时可以用 JAVA 加锁操作；

每 1 秒激活该线程计时，COUNTTIME+1 操作。

可以用此线程与作业请求线程、进程调度保持同步；

(3) 作业请求的中断仿真

假设计算机用自己的计时系统，每 10 秒(s)查询一次外部是否有新作业的执行请求，在“并发作业请求文件”中判读是否有新进程请求运行。

◆设计一个作业请求查询线程类，名称：JobIn_thread；

◆作业请求判读函数，名称：CheckJob（）：

功能：每 10 秒激活该线程。利用 COUNTTIME 变量每计时 10 秒，激活该线程，判断“并发作业请求文件”是否有新进程请求的时间已到达；如果有，进入后备队列；如果没有，空操作；

◆可以通过设置开关变量，通过时钟中断线程 Clock_thread 计时激活该线程；

◆也可以利用该线程的计时器计时激活该线程，但要利用 Clock_thread 线程的 COUNTTIME 变量判断，保证与系统时钟同步；

(4) 进程调度的中断仿真

假设计算机 CPU 在 1 秒(s)发生一次时钟硬件中断：可以执行 1 条指令。

假设进程调度的基础算法均为时间轮转法，在此基础上可以叠加静态或者动态特点，或者叠加静态或者动态优先权法；或者叠加多级反馈调度算法；

◆初始静态时间片大小：Times=3 秒

◆设计一个进程调度算法线程类，名称：ProcessScheduling_thread

线程激活时长：1 秒，也就是 1 秒发生一次调度中断信号，每次 Times-1；

◆可以通过设置开关变量，通过时钟中断线程 Clock_thread 计时激活该线程；

◆也可以利用该线程的计时器计时激活该线程，但要利用 Clock_thread 线程的 COUNTTIME 变量判断，保证与系统时钟同步；

◆调度时长设置函数，名称：SetSchedulingTime（）：

功能：以初始静态时间片大小为基础单元，根据选题设计具体对时间片大小 Times

重新赋值，当 Times=0 时发生进程调度；

- ◆进程调度函数，名称：ProcessScheduling（）：
功能：根据所选题目的调度算法实现调度；

（5）输入/输出中断事件仿真

仿真 I/O 中断。分别设计两个线程

- ◆InputBlock_thread 类：阻塞队列的唤醒线程。外设输入变量造成进程进入阻塞态，如键盘给变量赋值；
- ◆OutputBlock_thread 类：阻塞队列的唤醒线程。外设输出变量造成进程进入阻塞态，如显示器显示；

（6）用户区内存

假设不考虑 OS 内核所占内存空间

- ◆用户区共 16KB，每个物理块大小 1024B，共 16 个物理块；
- ◆用位示图实现内存管理。可以假设基础存储单元是 100B，每个物理块占用 10 个基础存储单元。
- ◆假设每条“用户态计算操作语句或者函数”类型的指令占用 100B，在进程创建时动态计算所创建进程所占用内存大小。每个进程的逻辑地址、物理地址均连续，物理地址的基地址可以不同；
- ◆地址变换过程(MMU)需要在界面上显示并详细记录过程，保存到 ProcessResults-???-算法名称代号.txt 文件中。??? 表示数字，为每次运行完成所有进程的总分钟数。

例如：测试输入数据是 input1 中的数据文件，用 LZ（时间片轮转）算法完成机器调度所有作业，总运行分钟数是 166 秒，保存进程运行输入输出以及调度过程中状态变化、统计数据等所有信息，保存到 ProcessResults-166-LZ.txt 文件，保存到 output1 子文件夹。

（7）内存缓冲区

- ◆内存缓冲区共 4KB，单缓冲，每个物理块大小 1024B，共 4 个物理块；
- ◆所有进程共享，进程通信过程中可能会出现死锁；

2.2 并发作业请求文件（jobs-input.txt）的设计

- ◆每个作业相当于用户提交给操作系统运行的可执行程序。本实验考虑作业调度、内存分配回收，进程管理等功能；

◆创建 input1、input2 子文件夹，保存 jobs-input.txt 以及用户程序指令文件。本次实验有两组测试输入数据，分别放在 input1、input2 子文件中。

input1 子文件：申请 C 及以下成绩的同学使用；

Input2 子文件：申请 B 及以上成绩的同学使用；

◆jobs-input.txt 文件格式固定，由教师统一提供，初始 5 个作业。一行代表一个作业请求，学生测试程序时可以增加新作业条目；

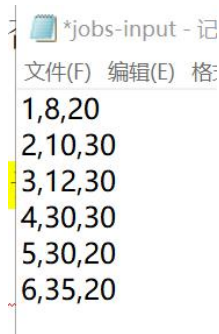
- ◆jobs-input.txt 文件内容：
 - 作业序号（JobsID）
 - 作业请求时间（InTimes）
 - 作业包含的程序指令数目（InstrucNum）

- ◆仿真程序开始运行时，开始按秒计时（不要用系统时间，用自己的计时器计时）。将

jobs-input.txt 文件一次性读入一个临时数组。**JobInput_thread** 类每 10 秒(s)查询临时数组, 根据当前计时器时间, 通过判断与作业请求时间 (**InTimes**) (单位为秒) 中请求时间的匹配程度, 判断该时刻是否有作业请求。如果有, 加入**作业后备队列**, 并根据作业编号找到该作业程序指令文件 (与 jobs-input.txt 文件在同一目录), 并将其内容保存到相应数据结构中。

◆在界面上需要设计一个**作业实时请求接收按钮**, 随时接收该时刻产生的新作业并加入到作业后备队列;

◆jobs-input.txt 文件举例:



2.3 用户程序指令文件的设计

◆用户程序指令文件是每个作业要执行的被编译以后的程序指令集合。每个作业的命令类型由本文档给出。

(1) 作业程序的结构

作业程序包括 **4** 类, 有语句和函数类型, 统一保存在 input1 和 input2 子文件夹中。每个用户程序指令文件名用“作业序号 (JobsID)”来命名。具体包括:

指令段编号 (**Instruc_ID**)

用户程序指令的类型 (**Instruc_State**)

(2) 用户程序指令的类型 (**Instruc_State**)

★本试验假设 CPU 执行用户态每段指令, 在执行一段机器指令时不可以中断。

★指令段编号 (**Instruc_ID**): 作业创建进程以后, 进程所执行用户程序段指令序号, 从 1 开始计数。

★根据 **Instruc_State** 类型, 每条指令运行或者唤醒时间 (**InRunTimes**) 有确定值, 不需要输入;

★用户程序指令的类型 (**Instruc_State**) 包括:

0 表示用户态计算操作语句。执行该指令需要运行时间 **InRunTimes=1s**。

1 表示用户态计算操作函数。执行该指令进程转向执行函数指令, 需要 **InRunTimes=2s**。

函数执行不可拆分;

当剩余时间片 < **InRunTimes**, 相当于时间片到; 如果 CPU 执行该计算操作函数时, 有优先级高的抢占进程请求, 则执行完该指令以后被抢占;

2 表示键盘输入变量语句。发生系统调用, CPU 进行模式切换, 运行进程进入阻塞态; 值守的键盘操作输入模块接收到输入变量或输出变量内容, **InRunTimes=2s** 后完成输入, 产生硬件终端信息号, **阻塞队列 1** 的队头节点出队, 进入就绪队列; **InputBlock_thread** 类在 **2s** 以后自动唤醒该进程; 会通过缓冲区进行数据输入输出;

3 屏幕输出显示语句。发生系统调用, CPU 进行模式切换, 运行进程进入阻塞态; 值

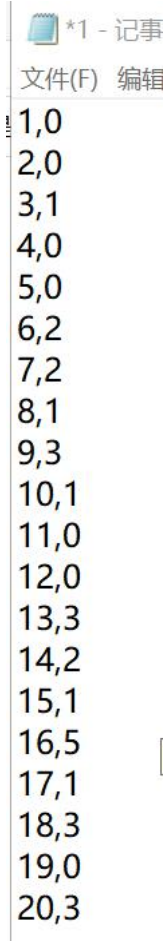
守的屏幕显示模块输出变量内容，InRunTimes=2s 后完成显示，产生硬件终端信息号，阻塞队列 2 的队头节点出队，进入就绪队列；OutputBlock_thread 类在 2s 以后自动唤醒该进程；

★如果按界面上实时作业请求按钮：可以随机生成用户程序指令文件；

举例：

选题 1：input1 文件夹中 1 号作业的程序段指令保存在 1.txt 文件中。2 号、3 号用户作业指令段保存在 2.txt、3.txt 文件中。文件第一行为代号。

1.txt 文件内容如下：



```

1,0
2,0
3,1
4,0
5,0
6,2
7,2
8,1
9,3
10,1
11,0
12,0
13,3
14,2
15,1
16,5
17,1
18,3
19,0
20,3

```

2.4 作业运行及调度详细记录的文件保存

◆作业和进程调度模块用单独的线程或计时器实现。需要重点注意的是：不是每次发生时钟中断都要进行进程调度。

◆时间片计算：进程调度最小时间片 3s。

◆进程调度算法代号：时间片轮转调度算法代号 SJP, 多级反馈队列调度算法代号 DJFK。

◆程序要求界面可视化显示作业请求与进程调度过程。同时将调度过程以文件形式保存。界面显示和文件保存的信息要清晰、完整、可读；

◆作业调度、进程调度运行完成以后，运行记录保存到 ProcessResults-???-算法名称代号.txt 文件。??? 表示数字，为每次运行完成所有进程的总分钟数。例如：测试输入数据是 input1 中的数据文件，用 SJP（时间片轮转）算法完成机器调度所有作业，总运行分钟数是 166 秒，保存进程运行输入输出以及调度过程中状态变化、统计数据等所有信息，保存到 ProcessResults-166-SJP.txt 文件，该保存到 output1 子文件夹。

保存内容至少包括：进程 ID、作业到达时间、作业运行时间、作业结束时间、作业中每条指令 ID、指令内容、指令状态转换的时间、CPU 寄存器变化情况、进程 PCB 信息等状态变化信息以及进程调度的周转时间等评价类数据。本次实验与 input1、input2 子文件测试数据对应的输出文件夹分别保存 output1、output2。该文件设计需要强调可读性与可分析性。

◆ProcessResults-???-算法名称代号.txt 文件内容的格式如下：

(1) 输出包括 4 段信息，分别为：进程调度事件；内存缓冲区处理事件；状态统计信息。其中，进程调度事件、状态统计信息是必须的，其他事件的记录需要根据选题及申请成绩等级来完成。

(2) 进程调度事件信息段的输出信息，每行有时间信息，按照时间顺序，记录发生进程调度事件、进程就绪队列、进程阻塞队列等的状态。

(3) 每段输出信息的关键信息加半角中括号[]。批改时会对所有带有中括号的内容进行提取，不加中括号都视为无效信息。

每一行信息的具体格式如下：

时间:[关键操作或状态名称：作业或进程信息，]

时间需要用仿真计时器的时间；

输出信息格式如下：

作业/进程调度事件：

0:[新增作业:作业编号,请求时间,指令数量]

1:[创建进程:进程 ID,PCB 内存块始地址,内存分配方式]

2:[进入就绪队列:进程 ID:待执行的指令数]

3:[运行进程:进程 ID:指令段编号,指令类型编号,逻辑地址,物理地址]

4:[运行进程:进程 ID:指令段编号,指令类型编号,逻辑地址,物理地址]

5:[运行进程:进程 ID:指令段编号,指令类型编号,逻辑地址,物理地址]

6:[阻塞进程:阻塞队列编号,进程 ID]

7:[缺页中断:进程 ID:PageOut 页号,PageIn 页号]

8:[重新进入就绪队列:进程 ID,待执行的指令数]

9:[阻塞进程:阻塞队列编号,进程 ID]

10:[CPU 空闲]

11:[终止进程 ID]

缓冲区处理事件：

3:[P 操作/V 操作+同步信号/同步信号]说明：2 段动作选择之一，组合显示并记录

4:[拷贝入缓冲区：进程 ID：指令段编号，磁盘文件读操作：逻辑地址，物理地址，缓冲区地址，交换区物理块地址]

5:[拷贝入缓冲区：进程 ID：指令段编号，磁盘文件读操作：逻辑地址，物理地址，缓冲区地址，交换区物理块地址]

6:[P 操作/V 操作+同步信号/同步信号]说明：2 段动作选择之一，组合显示并记录

7:[P 操作/V 操作+同步信号/同步信号]说明：2 段动作选择之一，组合显示并记录

8:[拷贝出缓冲区：进程 ID：指令段编号，磁盘文件读操作：逻辑地址，物理地址，缓冲区地址，交换区物理块地址]

9:[P 操作/V 操作+同步信号/同步信号]说明：2 段动作选择之一，组合显示并记录

10:[缓冲区无进程]

状态统计信息：

结束时间:[进程 ID:作业请求时间+进入时间+总运行时间]

结束时间:[进程 ID:作业请求时间+进入时间+总运行时间]

BB1:[阻塞队列 1,键盘输入:进入时间,进程 ID/进入时间,进程 ID/进入时间,进程 ID]

BB2:[阻塞队列 2,屏幕显示:进入时间,进程 ID/进入时间,进程 ID/进入时间,进程 ID]

说明：

在“作业/进程调度事件”中,f 符号均为半角西文态符号：

时间：每秒记录一条信息，两条记录的时间可以相同，表示并发操作；

关键操作或状态：包括新增作业、创建进程、第一次进入就绪队列、重新进入就绪队列阻塞队列、缺页中断、重新进入就绪队列、运行进程、CPU 空闲、终止进程等，输出时请写汉字信息；

记录事件的多少及时间根据选题确定；

作业或进程信息：包括进程 ID，指令编号，指令类型编号等，具体信息见下面的例子，信息用逗号隔开；

在“状态统计信息”中：当每个进程指令全部执行完成以后计算保存。每一行统计一个进程信息；

结束时间：指进程运行结束后的时间；

案例：只考虑进程调度事件、状态统计信息截图（格式与本要求不同）如下，仅供参考：

```
ProcessResults: 16.3-117K - 10/10/23
142:[运行进程:1,2,3,3902,1654]
142:[阻塞进程:OutputBlock_thread,1]
143:[缺页中断:3,1,9]
143:[运行进程:3,2,1,2,10001,4881]
143:[阻塞进程:inputBlock_thread,3]
144:[CPU空闲]
145:[重新进入就绪队列:3,9]
145:[重新进入就绪队列:1,2]
145:[CPU空闲]
146:[缺页中断:3,1,10]
146:[运行进程:3,2,2,1,11001,761]
147:[运行进程:3,2,2,0,11001,761]
148:[运行进程:3,2,3,1,10001,4881]
149:[运行进程:3,2,3,0,10001,4881]
150:[缺页中断:3,1,2]
150:[运行进程:3,2,4,0,2789,1765]
151:[重新进入就绪队列:3,6]
151:[缺页中断:3,1,3]
151:[运行进程:3,2,5,0,3290,2266]
152:[运行进程:3,2,6,1,10100,4980]
153:[运行进程:3,2,6,0,10100,4980]
154:[缺页中断:3,9,14]
154:[运行进程:3,2,7,1,14900,4660]
155:[运行进程:3,2,7,0,14900,4660]
156:[重新进入就绪队列:3,3]
156:[运行进程:3,2,8,0,3900,2876]
157:[缺页中断:3,10,5]
157:[运行进程:3,2,9,0,5789,669]
158:[缺页中断:3,2,4]
158:[运行进程:3,30,3,5000,1928]
158:[阻塞进程:OutputBlock_thread,3]
159:[缺页中断:1,1,1]
159:[运行进程:1,19,0,1610,4682]
160:[运行进程:1,20,3,4120,1048]
160:[阻塞进程:OutputBlock_thread,1]
161:[重新进入就绪队列:3,0]
161:[终止进程:3]
161:[CPU空闲]
162:[CPU空闲]
163:[重新进入就绪队列:1,0]
163:[终止进程:1]

状态统计信息:
163:[10+10+154]
105:[20+20+86]
161:[20+20+142]
109:[40+43+80]
BBBB:[OutputBlock_thread:16,1,18,1,64,4,65,3,67,4,68,3,69,4,86,3,105,3,134,1,135,3,138,1,139,3,143,3,
BBBB:[OutputBlock_thread:42,2,66,2,70,2,85,4,93,4,106,4,121,3,126,3,130,1,131,3,142,1,158,3,160,1,
PPPP:[PageInterruption_thread:10,1,12,1,14,1,15,1,18,1,20,2,22,2,24,2,30,2,32,2,36,2,42,2,43,4,44,4,45,4]
```

2.6 进程控制块 PCB 设计

下面是进程相关 PCB 内容，本次课设需要补充与逻辑地址、物理地址、外存物理块相关信息。参照 Linux task_struct 的数据结构内容设计；

进程编号（ProID）；值分别为 1, 2, 3, 4, 5, 6, 。。。

进程优先数（Priority）；

进程创建时间（InTimes）；

进程结束时间（EndTimes）；

进程状态（PSW）；

进程运行时间列表（RunTimes）；

进程周转时间统计（TurnTimes）；

进程包含的指令数目 (InstrucNum) ;
程序计数器信息 (PC) ;
指令寄存器信息 (IR) ;
在就绪队列信息列表 (包括: 位置编号 (RqNum)、进入就绪队列时间 (RqTimes)) ;
阻塞队列信息列表 1 (包括: 位置编号 (BqNum1)、进程进入键盘输入阻塞队列时间 (BqTimes1)) ;
阻塞队列信息列表 2 (包括: 位置编号 (BqNum2)、进程进入显示器输出阻塞队列时间 (BqTimes2)) ;

说明:

- ★系统请求运行的并发进程个数最小值为 5, 可以自行设计最大值。
- ★进程编号 (ProID): 整数
- ★进程优先数 (Priority): 随机生成[1-5]整数优先数, 优先数越小, 优先级越大;
- ★进程创建时间 (InTimes): 由仿真时钟开始计时, 整数, 假设每条指令执行时间 1s;
- ★进程结束时间 (EndTimes): 显示仿真时钟的时间, 整数;
- ★进程运行时间列表 (RunTime): 统计记录进程开始运行时间、时长, 时间由仿真时钟提供;
- ★进程包含的指令数目 (InstrucNum): 可自行扩展;
- ★PSW: 保存该进程当前状态: 运行、就绪、阻塞;
- ★指令寄存器信息 (IR): 正在执行的指令编号;
- ★程序计数器信息 (PC): 下一条将执行的指令编号;

2.7 系统 PCB 表的设计

需要设计系统空白 PCB 表, **假设系统最大 10 个进程并发**, 可用数组、链表、队列实现。

2.8 仿真实现进程控制原语

按照教材步骤仿真实现原语函数, 可以作为 PCB 类的方法, 包括:

进程创建: 考虑内存分配与回收。

进程撤销: 执行完成的进程调用撤销函数;

进程阻塞,I/O 操作产生, 可以不考虑对内存缓冲区的 PV 操作

2.9 其他

◆ **参加课程设计学生 2-4 人组成一个讨论小组, 1 位同学担任组长**, 负责组织集体讨论设计内容、组内测试。讨论组活动需要记录, 每次讨论, 每位同学需提供讨论报告。

◆本次课设每个讨论组活动次数不少于 5 次。讨论组由选题相同学生自由组合, 名单报本班学委助教。

◆每位同学独立完成本次课程设计内容, 独立完成课程设计与编程实现, 独立撰写实践报告, 独立答辩。实践报告中设计论述、问题分析记录是重点。

◆课程设计采用的语言为 JAVA 或 C#, 需要开发环境下运行以及脱离开发环境独立测试运行。

3 2023-2024 年第一学期课程设计时间节点安排

3.1 第 1 周（8 月 28 日-9 月 1 日），课程设计阅读指导手册，组建讨论组并开展活动；参考教材规划自己的课设计划，

3.2 第 2 周（9 月 5 日），周二晚上 6:00-7:00，课程设计线下辅导讲解（地点另行通知）

3.4 第 4 周（9 月 20 日，周三晚上 7:00-8:00），课设线上答疑，其他时间修改课设程序

3.5 第 5 周（9 月 27 日，周三晚上 7:00-8:00），课设线上答疑，其他时间修测试课设程序

3.6 第 7 周（10 月 9 日-13 日，全周停课课设, 地点：浦口校区实验室（地点另行通知）

3.7 正式提交全部材料时间：

10 月 12 日（周四）提交自己的课设完整材料：

10 月 13 日（周五）提交互测完整材料

3.8 停课 5 天时间安排：

停课课设期间每日点名，提交工作报告，作为平时成绩。具体安排以后续通知为准：

- （1）第 1-2 天：完善程序功能、修改可运行程序及代码，代码测试等；
- （2）第 3 天：发测试报告模板文件，完成程序代码测试，撰写测试报告，录制测试视频文件；
- （3）第 4 天：完成程序代码测试，完成自测报告、实践报告，录制带讲解的视屏文件等；提交自己参加课设所有正式材料；
- （4）第 5 天：学生相互测试，填写互测报告；提交互测所有正式材料。
- （5）因特殊原因不能到校上课的学生，需要提前一周提供学工办、教务办签字的证明，按照课设时间节点提交材料。
- （6）第 8-12 周，完成课设内容进一步测试与检查
- （7）第 13-15 周，完成抽查答辩及成绩统计
- （8）本学期课设采用新的自测报告模板、互测评价报告模板

4 正式材料提交要求

- （1）提交时间：2023 年 10 月 13 日（周五）16:00 前，只提交一次，过期不收。
- （2）要求学生通过提交完整全面的可执行程序、测序代码、测试报告、自我演示说明、实践报告等说服考评小组的材料。

◆提交材料的格式正确性、完整性与丰富程度，将占总成绩 10%

- （3）创建“姓名学号-课程设计-申请成绩”文件夹，进一步创建 input1、input2、output1、output2、code、doc、code-vidio、test-vidio、other 子文件夹（发的课设材料中已创建好了）。
- （4）“姓名学号-课程设计-申请成绩”文件夹的根目录：保存课设程序被编译为 RunProcess.exe 可执行文件；保存一份“材料阅读使用顺序说明文档”，给测试人、考评小组阅读提交材料的顺序及注意事项。

（3）input1、input2 子文件夹：

保存两组作业进程请求文件，申请 B 以下成绩的同学使用 input1 子文件夹测试输入数据；

申请 B 以上成绩的同学，使用 input2 子文件夹测试输入数据。每个子文件夹包括 jobs-input.txt、用户程序指令文件（如 1.txt 或 2.txt 等），由教师给定；

（4）output1、output2 子文件夹：分别对应 input1、input2 子文件夹输入作业的 ProcessResults-???-算法名称代号.txt 文件。

（5）code 子文件夹：工程文件、源程序等，要求公共变量、函数、函数输入输出变量有详细的注释，并提供本文件夹每个文件的说明视频文件；

（6）doc 子文件夹：保存提交的文档清单内容，全部是.docx 文件，每个文件名使用“编号-文件主题”命名。例如：1-课程设计实践报告；2-自测报告；3-程序安装及使用手册；4-讨论组讨论报告等

1-课程设计实践报告(电子版)

◆根据本文档“6. 课程设计实践报告内容与要求”写作；

2-自测报告（电子版）

说明：

◆根据自测报告模板要求完成，停课期间发；

◆自测报告需要提供分解以后具体的测试用例，至少包括测试功能名称、测试数据、测试输入输出结果界面、并对测试结果简要分析说明等；

◆根据课程设计评价标准说明的“百分比例”给每部分打分

◆给出自我评价的总成绩

◆每部分及总成绩等级分别为：A+、A、A-、B+、B、C、D、E

3-程序安装及使用手册（电子稿。包括：安装环境、方法、每个功能如何使用的例子）

4-讨论组讨论报告

说明：

◆每个人参与讨论组活动，每个讨论组由组长负责活动规划。

◆每次讨论需要主题。每一次都要写一个讨论报告。讨论报告是一种报告的形式，主写的是对讨论主题的分析论述语心得，每次讨论报告至少 300 字。次数不低于 3 次。讨论报告须包括：讨论时间、地点、参加人员、讨论时长、讨论内容分析、讨论心得等以及图片证明材料等。

（7）code-vidio 子文件夹：保存系统结构、每个实现代码讲解文件（多个），考核重点：

◆必须包括系统结构讲解文件。系统介绍整个课设的整体程序结构、每个模块、类、函数等代码结构与功能。突出讲解基础环节、重点环节、有特色环节的代码实现讲解；

◆按照申请成绩等级讲解以下仿实现代代码模块，每个模块单独录制视频文件。

◆单个文件大小最好不超过 30M

◆每个文件名用下面名称作为文件名，即使用“编号-实现主题”命名。

例如：1-工程文件及代码文件整体结构讲解；2-硬件设计讲解；等等

◆申请合格成绩的学生需要完成以下设计与实现的讲解（共 8 个文件）：

1-工程文件及代码文件整体结构讲解。包括讲解每个文件作用及相互关系；

2-硬件设计讲解

- 3-时钟中断实现代码讲解
 - 4-多线程检测作业并发请求；
 - 5-多线程实现作业调度；
 - 6-进程创建、撤销等原语；
 - 7-多线程实现时间片轮转调度算法（申 B 以上成绩的学生不用完成）；
 - 8-按要求格式读写输入、输出文件的实现过程；
 - ◆申请中等成绩的学生,还需要完成以下设计与实现代码的讲解(共 8+2=10 个文件):
 - 9-连续动态空间分配过程与回收过程的程序讲解；（申 A 以上成绩的学生不用完成）；
 - 10-可视化界面实现过程讲解；
 - ◆申请良好等级成绩的学生还需要完成以下设计与实现代码的讲解（共 7+2+4=13 个文件）：
 - 11-进程阻塞原语及调度过程代码讲解；
 - 12-进程唤醒原语及调度过程代码讲解；
 - 13-多级反馈队列调度算法代码讲解（申 A 以上成绩的学生不用完成）；
 - 14-接收界面上实时作业生成并参与调度的功能代码讲解；
 - ◆希望获得优秀成绩的学生还需要完成以下设计与实现的讲解（共 7+2+3+4=16 个文件）：
 - 15-内存缓冲区读写操作的实现代码讲解；
 - 16-伙伴系统的仿真实现代码讲解；
 - 17-死锁现象仿真的测试数据设计与实现过程讲解；
 - 18-死锁检测程序代码讲解；
- (8) test-vidio 子文件夹:对应输入输出文件以及保存每个用例测试过程的录屏讲解文件（可以多个）。这项内容将成为考核重要指标。
- ◆讲解以下测试用例，每个测试用例的测试过程单独保存录屏讲解文件：
 - ◆每个文件名使用“编号-测试主题”命名。例如：1-整体运行测试讲解
 - ◆单个文件大小最好不超过 30M；
 - ◆申请合格成绩的学生需要完成以下测试模块的讲解（共 3 个文件）：
 - 1-整体运行测试讲解。对应输入数据文件，讲解完整的运行调度过程，同时讲解 ProcessResults-???-算法名称代号.txt 文件生成过程，并对该文件中每条记录讲解；
 - 2-作业并发请求检测的测试讲解；
 - 3-不包括阻塞唤醒的多进程并发调度过程测试（申 B 以上成绩的学生不用完成）；
 - ◆申请中等成绩的学生还需要补充完成以下测试模块的讲解（共 3+1=4 个文件）：
 - 4-设置断点跟踪测试连续动态空间分配与回收过程的讲解；
 - ◆申请良好成绩的学生还需要补充完成以下测试模块的讲解（共 2+1+3=6 个文件）：
 - 5-设置断点跟踪测试多级反馈队列调度算法（申 A 以上成绩的学生不用完成）；
 - 6-设置断点跟踪测试阻塞唤醒过程；
 - 7-实时作业生成并参与调度测试讲解；
 - ◆申请优秀成绩的学生还需要补充完成以下测试模块的讲解（共 3+1+2+3=9 个文件）
 - 8-设置断点跟踪内存缓冲区读写操作功能测试讲解；
 - 9-设置断点跟踪伙伴系统的功能测试讲解；
 - 10-设置断点跟踪死锁现象仿真的测试数据与死锁检测模块测试讲解；

(9) other 文件夹：保存使用了第三方程序或框架的库文件、安装使用手册等

5 课程设计评价标准说明

5.1 成绩百分比例

通过答辩讲解、测试程序、观看视频、阅读实践报告等方式评价。

◆提交材料的格式正确性、完整性与丰富程度，将占总成绩（10%）

◆机器自动批改：占总成绩 20%

◆代码测试：占总成绩 30%

◆报告、视频讲解及答辩：占总成绩 40%

◆平时成绩：占总成绩 10%

5.2 基础性评价标准

◆按照关键点得分；

◆必须完成本实践要求第 1 节的基础要求；

◆在进程相关 PCB 内容基础上，本次课设必须补充与逻辑地址、物理地址、外存物理块相关 PCB 信息

◆可以用 Java 或 C#编程实现；

◆必须按照“用户程序段指令文件”、ProcessResults-??.txt 文件格式要求读入或输出

◆必须实现时间中断仿真

◆必须实现作业及进程并发请求功能，必须用至少 3 个线程编程实现；

◆必须实现对作业或进程的指令段集调度功能；

◆必须提供 3 种以上用户程序场景的测试，例如：键盘、显示器输入输出和计算的程序等；

◆代码查重相似度超过 85%以上为疑似抄袭，认定抄袭后按 0 分计并报教务处；

◆依据《计算机操作系统》原理实现仿真功能，设计、实现及答辩与理论差距过大或者没有理论依据，不得分；

◆先仿真《计算机操作系统》内核功能，可视化仿真管理过程属于加分项；

◆仿真 OS 内核的代码量不能少于 800 行；

◆成绩水平分数范围见下表：

成绩水平	成绩范围（百分数）
不能执行	0
基础要求没有实现	20-40
基础要求功能实现不充分或不完整	50-60
令人满意（在正确性和完整性方面）	60-80
非常好（就完整性和鲁棒性而言）	80-90
出色到完美（在完整性，健壮性和复杂性方面）	90-100

6 课程设计实践报告内容与要求

注意：逻辑结构合理，层次分明，论述清楚，不要放程序代码，核心算法与关键环节处理用伪码描述。对每个测试结果需要分析说明。参考文献（5 篇以上，采用南农自然科学版格式排版）

正文：宋体 5 号字，图表需要有编号和名称

封皮（统一格式可下载）

目录

摘要（单独一页，500 字以上，包括目的、完成功能、总结核心设计实现技术、测试结果总结等，需要具体化）

关键词（不超过 5 个，与摘要在一页）

1 实践内容与目标

包括：1.1 实践内容、1.2 实践目标（分条目论述完成了任务、实现函数、模块功能）

2 系统功能与 API 函数说明

包括：描述所设计的系统功能说明，所对应的 API 函数名称；

API 函数需要说明函数输入、输出变量说明及调用方法

3 硬件仿真设计

分小标题论述，包括：计算机各硬件部件的仿真设计，要有具体的大小和仿真方法等

4 数据结构与基础操作的抽象与实现

分小标题论述，包括：作业控制块及程序指令集设计、与硬件映射的数据结构设计、OS 原语与核心操作，采用示意图与伪码相结合的方式，不能直接拷贝代码等，注意写清楚自己有特色的关键处理；

5 程序整体结构及模块功能的实现

分小标题论述，包括：整个软件系统的结构说明，采用示意图、流程图与伪码相结合的方式说明，不能直接拷贝代码。注意写清楚自己有特色的关键处理；

6 技术问题及解决方案（重点）

分小标题论述，要有具体技术问题描述，发生场景描述，最好配合现场图片；给出具体解决方案和结果测试截图证据等

7 实践心得（具体化）

参考文献（5 篇以上，采用南农自然科学版格式排版）

附件 1：程序文件及结构说明

附件 2：类图和顺序图说明

附件 3：带注释的部分核心代码（只需要提交 5 页）

7 课程设计课程联系方式

设计问题可通过 QQ 群或当面交流联系任课教师；

办公地点：育贤楼 B303