

华东师范大学期中考试试卷
2019 —2020 学年第一学期

课程名称：___操作系统___

学生姓名：_____

学 号：_____

专 业：_____

年级/班级：_____

课程性质：公共必修、公共选修、专业必修、专业选修

一	二	三	四	五	六	七	八	总分	阅卷人签名

一、判断题（20 分，每小题 4 分）

判断下列每句话是否正确，如错误请说明理由。

1. 时间片轮转调度算法、短进程优先调度算法、先来先服务调度算法, 及高响应比优先调度算法都没有综合考虑进程等待时间和执行时间。 (X)

答案：错。高响应比优先调度算法综合考虑进程等待时间和执行时间

- ~~2. 多级反馈队列的调度中就绪队列按时间片的大小划分。 (对)~~

2. 多级反馈队列中，进程可以在不同队列中移动 (√)

3. 在生产者-消费者问题中，生产者进程和消费者进程只要解决互斥访问的问题即可。

答案：错。当 buffer 为空时，只有等待生成者进程完成一个产品生成后，消费者才可以消费。因此，还要解决进程间同步的问题。

4. 进程执行的相对速度与资源无关。 (X)

答案：错。 进程执行的相对速度与资源有关，例如一个进程执行时需要的资源不能满足要求时，只好等待该资源。

- ~~5. 在多处理器系统中，通过关中断的方式可以解决进程间互斥访问的问题。~~

答案：错。在多处理器系统中，关中断不能保重临界区的互斥访问。

5.在微内核（micro-kernel）结构的操作系统中，虚拟内存（virtual memory）管理是在微内核内部的。

错。只有 CPU 调度和进程间通信是必须在微内核内部。

二、不定项选择题（20 分，每小题 4 分）

每题有一个或多个答案，答错、少选、多选均不给分。

1. 当 CPU 执行操作系统代码时,处理器处于（ D ）
A. 执行态 B. 用户态 C. 就绪态 D. 内核态
2. UNIX 操作系统是典型的（ B ）
A. 多道批处理系统 B. 分时系统 C. 实时系统 D. 分布式系统
3. 以关于线程，以下说法正确的是（ A, B, C ）
A. 用户态线程（无核心态线程或 LWP）阻塞，可能会阻塞线程
B. 多处理器环境下，线程间同步不能使用关中断实现
C. 线程控制块中包含 CPU 寄存器状态
D. 在支持核心态线程的系统中，CPU 调度的单位仍然是进程
4. 一个正在运行的进程，当所分配的时间片用完后，将其挂在（ C ）。
A. 等待队列
B. 运行队列
C. 就绪队列
D. 任意一个队列
5. 若 P、V 操作的信号量 S 初值为 2，当前值为 -1，则表示有（ B ）等待进程。
A. 0 个 B. 1 个 C. 2 个 D. 3 个

三、辨析题（20 分，每小题 5 分）

分别解释以下每组的两个名词，并列举它们的区别和联系

1. 进程和管程
2. 多级反馈队列调度与 FCFS
3. 死锁（deadlock）与饥饿（starvation）

死锁：多个进程循环等待对方，都无法继续执行

饥饿：某个或某些进程由于无法得到资源长时间无法执行

死锁必然发生饥饿，但是饥饿不一定发生死锁

4. 互斥量和信号量

【参考】1) 互斥量用于线程的互斥，信号量用于线程的同步。

这是互斥量和信号量的根本区别，也就是互斥和同步之间的区别。

互斥：是指某一资源同时只允许一个访问者对其进行访问，具有唯一性和排它性。但互斥无法限制访问者对资源的访问顺序，即访问是无序的。

同步：是指在互斥的基础上（大多数情况），通过其它机制实现访问者对资源的有序访问。在大多数情况下，同步已经实现了互斥，特别是所有写入资源的情况必定是互斥的。少数情况是指可以允许多个访问者同时访问资源

2) 互斥量值只能为 0/1，信号量值可以为非负整数。

也就是说，一个互斥量只能用于一个资源的互斥访问，它不能实现多个资源的多线程互斥问题。信号量可以实现多个同类资源的多线程互斥和同步。当信号量为单值信号量是，也可以完成一个资源的互斥访问。

3) 互斥量的加锁和解锁必须由同一线程分别对应使用，信号量可以由一个线程释放，另一个线程得到。

四、综合题 （40 分）

1. 四个进程需要运行的时间（单位：毫秒）和到达顺序如下：（10 分）

进程	需要 CPU 时间	优先级
P1	9	3
P2	2	1
P3	3	2
P4	2	4

- a) 请利用甘特图展示四个进程分别在 FCFS、SJF、RR 和非抢占优先级四种调度算法下的执行过程。
- b) 请计算每个进程在 a) 中提到的四种调度算法下的等待时间。
- c) 请问 a) 中哪种调度算法使得平均等待时间最少？

(b)

进程	FCFS	SJF	RR	优先级调度
P1	0	7	6	5
P2	9	0	4	0
P3	11	4	7	2
P4	14	2	6	14
平均	8.5	3.25	5.6	5.2

(c) SJF 调度算法使得平均等待时间最少。

2. 请举例详细说明什么是进程之间的同步和互斥，为什么中断不适合在多处理器系统中实现原语的同步？（10 分）

参考答案：

互斥：是指某一资源同时只允许一个访问者对其进行访问，具有唯一性和排它性。但互斥无法限制访问者对资源的访问顺序，即访问是无序的。

同步：是指在互斥的基础上（大多数情况），通过其它机制实现访问者对资源的有序访问。在大多数情况下，同步已经实现了互斥，特别是所有写入资源的情况必定是互斥的。少数情况是指可以允许多个访问者同时访问资源

在实际的多道程序系统中，这种连续循环浪费了 CPU 时间。操作中信号量整数值的修改应当不可被分割地执行，且不能被中断，运行在处理器上的是内核级线程 而用户级程序所对应的内核线程是需要进行调度的，所以不确保禁止中断实现同步原语的方法有效。

3. “生产者-消费者”问题演变 2（既有同步，也有互斥）：一个缓冲区，多个生产者，多个消费者，生产者不断地生产，消费者不断地消费。只有缓冲区为空时生产者才能放产品，只有缓冲区有数据，消费者才能取产品。用 **PV** 操作写出相应的代码段。（10 分）

答案：缓冲区变成临界资源，不允许多个进程同时操作缓冲区，既不允许多个生产者同时放产品，也不允许多个消费者同时取产品。那么需要增加信号量 **mutex** 实现对缓冲区的互斥访问，初值为 1。
full 和 **empty** 范围分别变成 $[-n,1]$ 和 $[-m,1]$ ，其中 **n** 是消费者进程数量，**m** 是生产者进程数量。

Semaphore mutex=1,empty=1,full=0;

main()

{

// begin

Producer();

```

    Consumer();
// end
}
Producer()
{
    While(true)
    {
        wait(empty);
        wait(mutex)
        putdata;
        signal (mutex)
        signal(full);
    }
}

Consumer()
{
    While(true)
    {
        wait(full);
        wait(mutex)
        getdata;
        signal (mutex)
        signal(empty);
    }
}

```

4. 举例说明什么是多级反馈队列调度算法，其有哪些优点，根据例子具体详细说明。（10分）

参考答案：多级反馈队列调度算法是一种性能较好的作业低级调度策略，能够满足各类用户的需要。对于分时交互型短作业，系统通常可在第一队列（高优先级队列）规定的时间片内让其完成工作，使终端型用户都感到满意；对短的批处理作业，通常，只需在第一或第一、第二队列（中优先级队列）中各执行一个时间片就能完成工作，周转时间仍然很短；对长的批处理作业，它将依次在第一、第二、……，各个队列中获得时间片并运行，决不会出现得不到处理的情况。此系统模拟了多级反馈队列调度算法及其实现。