

武汉大学国际软件学院 2012-2013 学年第二学期期末考试试卷

课程名称:《操作系统》(B 卷)

专业: 软件工程

层次: 本科

年级: 2011

姓名: _____ 学号: _____ 考分: _____

说明: 1、答案一律书写在答题纸上, 书写在试卷上或其他地方一律无效。

2、请准确规范书写姓名和学号, 否则作废。

一、判断题(每空 1 分, 共 10 分)

1. 按优先级调度算法, 处于运行状态的进程一定是所有进程中优先级最高的进程。F
2. 并发性是指若干事件在同一时刻发生。F
3. 虚存容量的扩大是以牺牲 CPU 工作时间以及内、外存交换时间为代价的。T
4. 页式存储管理中, 一个作业可以占用不连续的内存空间, 而段式存储管理, 一个作业则是占用连续的内存空间。F
5. 对临界资源应采取互斥访问方式来实现共享。T
6. 页式的地址是一维的, 段式的地址是二维的。T
7. 系统发生死锁时, 其资源分配图中必然存在环路。因此, 如果资源分配图中存在环路, 则系统一定出现死锁。F
8. 一个正在运行的进程可以阻塞其他进程。但一个被阻塞的进程不能唤醒自己, 它只能等待别的进程唤醒它。F
9. 信箱通信是一种进程间直接通信方式。F
10. 顺序式文件结构不利于文件长度的动态增长。T

二、单选题(每空 2 分, 共 40 分)

1. 从多道批处理系统发展到分时系统的主要原因是 ()。
A. 前者系统吞吐量小 B. 进一步提高 CPU 利用率
C. 后者各类资源利用率更高 D. 后者能够提供交互能力
2. 下面哪种算法不是 LRU 算法的实现、或者近似实现算法? Tag!!
A. 简单时钟置换算法 B. 改进时钟置换算法
C. 最近使用置换算法 D. 基于 n 位移位寄存器的置换算法
3. 多道程序设计的关键问题不包括 ()。
A. 处理机调度 B. 内存管理 C. I/O 管理 D. 扩展计算机的原有功能
4. 根据银行家算法, 系统中有 4 个并发进程, 如果每个进程最大需要 3 个该类资源。试问该类资源最少为 () 个时, 此时肯定存在安全序列。
A. 9 B. 10 C. 11 D. 12
5. 采用资源有序分配算法可以 ()。 Tag!!
A. 预防死锁 B. 解除死锁 C. 避免死锁 D. 检测死锁

6. 在下列寻道优化算法中, 可能出现饥饿现象的是 ()。Tag!!
A. 先来先服务 B. 扫描算法 C. 最短寻道时间优先 D. 循环扫描算法
7. 进程间高级通信机制不包括 ()。
A. 信号量机制 B. 共享存储器机制 C. 消息通信机制 D. 管道机制
8. 把作业地址空间中使用的逻辑地址变成内存中的物理地址称为 ()。
A. 加载 B. 重定位 C. 物理化 D. 逻辑化
9. 一作业 8:00 到达系统, 估计运行时间为 1 小时。若 10:00 开始执行该作业, 则此时其响应比是 ()。
A. 2 B. 1 C. 3 D. 0.5
10. 联想存储器在计算机系统中是用于_____的。
A. 存储文件信息且 B. 与主存交换信息 C. 地址变换 D. 存储通道程序
11. 一个进程可以包含多个线程, 各线程____。
A. 必须串行工作 B. 共享分配给进程的主存地址空间
C. 共享进程的 PCB D. 是独立的资源分配单位
12. 采用 SPOOLing 技术的目的是 ()。
A. 提高独占设备的利用率 B. 提高主机效率
C. 减轻用户编程负担 D. 提高程序的运行速度
13. 虚拟存储器实现的基础是 () 理论。
A. 局部性 B. 全局性 C. 动态性 D. 虚拟性
14. 当采用单缓冲技术进行磁盘输入时, 设从磁盘上读入 1 块的时间为 T , 将数据从单缓冲区送入用户区所需时间为 t , 用户程序处理这块数据的时间为 p , 且 $T > t$, $T > p$ 。如果需从磁盘上读入 2 块数据, 并进行处理, 则总共需要花费的时间为 ()。
A. $2T+2t+2p$ B. $2T+2t+p$ C. $2T+t+2p$ D. $2T+t+p$
15. 可变分区存储管理的主存分配算法中, 寻找次数最少的是 ()。Tag!!
A. 首次适应算法 B. 循环首次适应算法
C. 最佳适应算法 D. 最坏适应算法
16. CPU 输出数据的速度远远高于打印机的打印速度, 为了解决这一矛盾, 可采用 ()。
A. 并行技术 B. 通道技术 C. 缓冲技术 D. 虚存技术
17. 在几种常见的 I/O 控制方式中, CPU 和外围设备只能串行工作的是 ()。
A. 程序直接控制方式 B. 中断方式 C. DMA 方式 D. 通道控制方式
18. 多个作业同时到达时, 下列作业调度算法中, 具有最短的作业平均周转时间的是 ()。
A. 先来先服务法 B. 短作业优先法 C. 优先数法 D. 时间片轮转法
19. 常用的文件存取方法有两种: 顺序存取和 () 存取。
A. 流式 B. 串联 C. 索引 D. 随机
20. 进程从运行状态到等待状态可能是由于 ()
A. 进程调度程序的调度 B. 现运行进程时间片用完
C. 现运行进程执行了 wait 操作 D. 现运行进程执行了 signal 操作

三、填空题（每空 1 分，共 10 分）

1. 在操作系统中，进程是一个 资源分配 的基本单位，也是一个独立运行和调度的基本单位。
2. 现代操作系统的两个基本特征是 并发、共享 和虚拟。
3. 若信号量 S 的初值定义为 10，则在 S 上调用了 16 次 wait 操作和 15 次 signal 操作后 S 的值应该为 9。
4. 在操作系统中，不可中断执行的操作称为 原语_原子操作。
5. 把作业装入内存中随即进行地址变换的方式称为静态地址重定位，而在作业执行期间，当访问到指令或数据时才进行地址变换的方式称为 动态地址重定位。
6. 文件系统中若文件的物理结构采用连续结构，则文件控制块中关于文件的物理位置应包括 起始地址 和总块数。
7. 进程通常由程序段、相关数据_数据段 和 PCB 三部分组成。
8. 在一分页存储管理系统中，页面大小为 1K 字节，作业 A 的 5 个页面 0、1、2、3、4 被分配到内存的 3、5、1、4、7 块中，则逻辑地址 3456 对应的页号为 4 3，物理地址为 4480。

四、简答题（每小题 4 分，共 12 分）

1. 什么是操作系统？从资源管理的角度说明操作系统的主要功能。
2. 什么是临界区？进程进入临界区的调度原则是什么？
3. 为什么要引入虚存的概念？虚存的最大容量由什么决定？
为了

五、计算题（每小题 6-8 分，共 28 分）

1. (7 分) 设某作业占有 7 个页面，如果在主存中只允许装入该作业的 4 个页面，作业运行时，实际访问页面的顺序是：1, 2, 3, 6, 4, 7, 3, 2, 1, 4, 7, 5, 6, 5, 2, 1。试用下述算法，列出各自的页面淘汰顺序和页面置换次数。
 - (1) FIFO 页面置换算法
 - (2) LRU 页面置换算法
 - (3) CLOCK 页面置换算法
2. (6 分) 若干个等待访问磁盘者依次要访问的磁道为 20, 44, 40, 4, 80, 12, 76，假设每移动一个磁道需要 3 毫秒时间，移动臂当前位于 40 号柱面，请按下列算法分别写出访问序列并计算为完成上述各次访问总共花费的寻道时间。
 - (1) 先来先服务算法；
 - (2) 最短寻道时间优先算法。
 - (3) 扫描算法（当前磁头移动的方向为磁道递增）
3. (7 分) 某系统有 A、B、C、D 四类资源可供五个进程 P1、P2、P3、P4、P5 共享。系统

对这四类资源的拥有量为:A 类 3 个、B 类 14 个、C 类 12 个、D 类 12 个。进程对资源的需求和分配情况如下:

进程	已占有资源				最大需求数			
	A	B	C	D	A	B	C	D
P1	0	0	1	2	0	0	1	2
P2	1	0	0	0	1	7	5	0
P3	1	3	5	4	2	3	5	6
P4	0	6	3	2	0	6	5	2
P5	0	0	1	4	0	6	5	6

按银行家算法回答下列问题:

- (1) 现在系统中的各类资源还剩余多少?
- (2) 现在系统是否处于安全状态? 为什么?
- (3) 如果现在进程 P2 提出需要 A 类资源 0 个、B 类资源 4 个、C 类资源 2 个和 D 类资源 0 个, 系统能否去满足它的请求? 请说明原因。

4. (8 分) 在读者写者问题中, 读者优先策略使得后到达的读者可能先于写者访问数据文件, 那么如何为写者进行改进, 而使其不会出现饥饿的现象呢? 这里增加了一个排序队列 queue, 所有读者写者都需在此队列排队, 然后依次访问数据文件, 即公平竞争。请填充写者程序中空白处的 wait 和 signal 操作。

Semaphore fmutex=1, rdcntmutex=1, queue=1; int readcount = 0;

//fmutex --> access to file; rdcntmutex --> access to readcount

<pre> void reader(){ while(1){ wait(queue); wait(rdcntmutex); if(0 == readcount)wait(fmutex); readcount = readcount + 1; signal(rdcntmutex); signal(queue); //Do read operation ... wait(rdcntmutex); readcount = readcount - 1; if(0 == readcount)signal(fmutex); signal(rdcntmutex); } } </pre>	<pre> void writer(){ while(1){ (1) _____; (2) _____; (3) _____; //Do write operation ... (4) _____; } } </pre>
---	--

