

座位号：

杭州电子科技大学学生考试卷（A）卷

考试课程	操作系统，操作系统原理		考试日期	2024年1月19日		成绩	
课程号	A0503030 A0505660	教师号		任课教师姓名		赵伟华, 刘真, 贾刚勇, 周旭, 王俊美, 顾人舒, 崔扬	
考生姓名		学号(8位)		年级		专业	

注意事项：用黑色字迹签字笔或钢笔将答案写在答题纸上，答题纸上写明学号和姓名。试卷和答题纸装订在一起上交。

一、选择题（每题1分，共25分）

1. 在中断发生后，进入中断处理的程序属于（ ）
A. 用户程序 B. 可能是应用程序，也可能是操作系统程序
C. 操作系统程序 D. 既不是应用程序，也不是操作系统程序
2. 以下（ ）指令是非特权指令。
A. 启动 I/O B. 设置中断屏蔽 C. 传送 PSW D. 访管指令
3. 有关原语的说法中，（ ）是正确的。
A. 原语是不可中断执行的用户过程
B. 原语是不可中断执行的操作系统过程
C. 原语是可中断执行的用户过程
D. 原语是可中断执行的操作系统过程
4. Linux 用于启动系统所需加载的内核程序位于（ ）
A. / B. /lib/modules/2.4.20_8/kernel C. /boot D. /proc
5. 下列选项中，不能改善磁盘 I/O 性能的是（ ）
A. 重排 I/O 请求次序 B. 在一个磁盘设置多个分区
C. 预读和滞后写 D. 优化文件物理分布
6. 为了防止用户共享文件时造成破坏，可以采用（ ）
A. 对文件设置口令 B. 把文件译成密码
C. 对文件加锁 D. 对文件的访问权限进行控制
7. 若采用位示图（100行，32列）表示磁盘块的使用状态。当分配一个盘块号133号时，其在位示图中的行、列数位（ ）。（注：行号：0-99，列为0-31，首盘块号为0）
A. 4和5 B. 5和3 C. 4和3 D. 5和4

8. 某文件系统的目录项由文件名和索引结点好构成。若每个目录项长度为64字节，其中4字节存放索引结点号，60字节存放文件名。文件名由小写英文字母构成，则该文件系统能创建的文件数量的上限为（ ）个。
A. 2^{26} B. 2^{32} C. 2^{60} D. 2^{64}
9. 以下几种内存管理方式中，只会形成外部碎片的管理方式是（ ）。
A. 单一连续分配方式 B. 固定分区分配方式
C. 段式存储管理方式 D. 段页式存储管理方式
10. 下面关于内存保护的界限寄存器方法的描述中，正确的是（ ）
A. 界限寄存器方法通常用在内存离散分配方式的保护机制中
B. 在上下界寄存器方法中，使用逻辑地址进行越界检查
C. 在基址和限长寄存器方法中，使用逻辑地址进行越界检查
D. 以上说法都不对
11. 在支持多线程的系统中，隶属于同一个进程的多个线程不能共享的是（ ）。
A. 进程的代码段 B. 进程的数据段
C. 进程所打开的文件 D. 保存函数参数、返回地址等信息的堆栈
12. 有一个100行×200列的矩阵，在一个虚拟存储系统中，采用LRU算法，系统分给该进程5个页面来存储数据（不包含程序），设每页存放200个整数，数组是按行存放的，下面程序要对数组进行初始化，则缺页次数为（ ）次。

```
for (i=0; i<=99; i++)
    for (j=0; j<=199; j++)
        A[i][j]=i+j;
```


A. 100 B. 200 C. 300 D. 20000
13. 在一个请求分页存储管理系统中，某时刻测得系统各相关设备的利用率为：CPU为12%，磁盘交换区为99.5%，其他I/O设备为5%，下面（ ）措施可以更有效地改进CPU的利用率？
A. 增大磁盘交换区的容量 B. 减少内存中程序道数
C. 使用更快速的CPU D. 增加内存中程序道数
14. 在如下几种类型的系统中，（ ）采用忙等待I/O是合适的。
I. 专门用来控制单I/O设备的系统
II. 单用户单任务操作系统
III. 作为一个负载很重的网络服务器的工作站
A. I B. II, III C. I, II D. I, II, III
15. 与Linux系统的整体式内核结构相比，采用微内核结构的鸿蒙操作系统具有的特征是（ ）。
I. 较高的效率; II. 较高的可靠性;
III. 更好地支持分布式处理; IV. 较强的可扩展性
A. II、IV B. I、II、III C. I、III、IV D. II、III、IV

<p>16. 某文件系统采用位示图管理文件存储空间，文件存储空间大小为 256GB，盘块大小为 4KB，则位示图所占空间大小是（ ）。</p> <p>A. 2MB B. 8MB C. 64MB D. 4KB</p> <p>17. 下列事件中，可能导致当前正在执行的线程由执行态转变为就绪态的是（ ）。</p> <p>A. 键盘输入 B. 缺页异常 C. 主动出让 CPU D. 执行信号量的 wait() 操作</p> <p>18. 对于采用虚拟内存管理方式的系统，下列关于进程虚拟地址空间的叙述中，错误的是（ ）。</p> <p>A. 每个进程都有自己独立的虚拟地址空间 B. C 语言中 malloc() 函数返回的是虚拟地址 C. 进程对数据段和代码段可以有不同的访问权限 D. 虚拟地址的大小由主存和硬盘的大小决定</p> <p>19. 若文件 F 仅被进程 P 打开并访问，则当进程 P 关闭 F 时，下列操作中，文件系统需要完成的是（ ）。</p> <p>A. 删除目录中文件 F 的目录项 B. 释放 F 的索引节点所占的内存空间 C. 释放 F 的索引节点所占的外存空间 D. 将文件磁盘索引节点中的链接计数减 1</p> <p>20. 引入多道程序技术的前提之一是系统具有（ ）。</p> <p>A. 多个 CPU B. 多个终端 C. 中断功能 D. 分时功能</p> <p>21. 在消息传递通信机制中，发送原语 send 要做的工作不包括（ ）。</p> <p>A. 在发送进程的内存空间中设置一个发送区，并填写相关信息 B. 在系统中申请一个空白消息缓冲区 C. 将发送区中的信息复制到消息缓冲区中 D. 将消息缓冲区插入接收进程的消息队列</p> <p>22. 两个进程 A 和 B，每个进程都需要使用打印机 a、扫描仪 b、绘图仪 c 三个资源。系统中每种资源只有 1 个。如果 A 和 B 都以 abc 的次序请求，则系统死锁无关；但是若 A 以 abc 的次序请求资源，B 以 cba 的次序请求资源，则有可能死锁。那么对于这 3 种资源的不同请求次序组合中，有（ ）几率可以保证不会发生死锁。</p> <p>A. 1/2 B. 1/3 C. 1/4 D. 1/6</p> <p>23. A high-level abstraction that provides a convenient and effective mechanism for process synchronization. Only one process may be active within it at a time.</p> <p>这段话描述的是（ ）。</p> <p>A. pipe B. monitor C. semaphore D. thread</p>	<p>24. 在 Linux 中运行下面框图中的程序，则所给出的输出结果中可能出现的有（ ）。</p> <p>I. abcc II. bcac III. abac IV. acbc V. cabc</p> <p>A. I, II, III B. I, II, IV C. II, IV, V D. II, III, IV</p> <pre>main(){ int x; while((x=fork())== -1); if(x==0) printf("a"); else printf("b"); printf("c"); }</pre> <p>25. 进程 P1 和 P2 都包含并发线程，伪代码描述如下：</p> <table border="1" data-bbox="1699 752 2810 1021"> <tr> <th>//进程 P1</th> <th>//进程 P2</th> </tr> <tr> <td>int x=0;</td> <td>int x=0;</td> </tr> <tr> <td>thread1() {</td> <td>thread2() {</td> <td>thread3() {</td> <td>thread4() {</td> </tr> <tr> <td> int a;</td> <td> int a;</td> <td> int a;</td> <td> int b;</td> </tr> <tr> <td> a = 1;</td> <td> a = 2;</td> <td> a = x;</td> <td> b = x;</td> </tr> <tr> <td> x += 1;</td> <td> x += 2;</td> <td> x += 3;</td> <td> x += 4;</td> </tr> <tr> <td>}</td> <td>}</td> <td>}</td> <td>}</td> </tr> </table> <p>下列选项中，需要互斥操作的是（ ）。</p> <p>A. a = 1 与 a = 2 B. a = x 与 b = x C. x += 1 与 x += 2 D. x += 1 与 x += 3</p>	//进程 P1	//进程 P2	int x=0;	int x=0;	thread1() {	thread2() {	thread3() {	thread4() {	int a;	int a;	int a;	int b;	a = 1;	a = 2;	a = x;	b = x;	x += 1;	x += 2;	x += 3;	x += 4;	}	}	}	}
//进程 P1	//进程 P2																								
int x=0;	int x=0;																								
thread1() {	thread2() {	thread3() {	thread4() {																						
int a;	int a;	int a;	int b;																						
a = 1;	a = 2;	a = x;	b = x;																						
x += 1;	x += 2;	x += 3;	x += 4;																						
}	}	}	}																						

座位号:

3. (12分) 某虚拟存储系统中有一个进程共有6页(0-5):代码占3页(页号为0,1,2),数据占1页(页号为3),数据堆占1页(页号为4),用户栈占1页(页号为5),它们依次存放在外存的22、23、25、26号磁盘块中。当前代码页分配在内存的66、67、87号块中;数据页分配在31号块中,并已经被修改;数据堆页还没有分配内存;用户栈分配在1号块中,未被修改。请完成以下问题:

(1) 完成下面页表的内容:

页号	块号	修改位	访问位	引用时间	外存地址	存在位
				1203		
				1178		
				1225		
				1020		
				—		
				1250		

(2) 若数据堆申请内存,因未分配物理内存而产生缺页中断,假设系统采用固定分配、局部置换策略,且采用LRU页面置换算法,则应淘汰哪个页面?操作系统如何处理?页表内容又如何变化?设当前时刻为虚拟时间1270。

4. (14分) 在一处很深的南北走向的非洲峡谷上,有一根坚固的横跨峡谷的绳索,狒狒可以攀住绳索越过峡谷。同一时刻,只要朝着相同的方向,可以若干只狒狒同时通过。但是向东和向西的狒狒同时攀在绳索上就无法通行,狒狒会被卡在中间,它们无法在绳索上从另一只的背上翻过去。

(1) 请利用信号量机制编写伪代码程序来解决该问题。

(2) 请分析上述通行规则可能存在什么问题?提出一种解决思路(无需伪代码)。

5. (13分) 某32位请求分页系统采用二级页表结构,外部页表项和进程页表项长度均为4字节,虚拟地址结构如下图所示:



某C程序中数组A[512][512]的起始虚拟地址为1080 0000H,按行优先方式连续存放在进程虚拟地址空间中,每个数组元素占4字节。外部页表在内存中的起始物理地址为

0020 1000H,请回答以下问题:

(1) 数组元素A[1][2]的虚拟地址是什么?对应的外部页号和外部页内地址分别是多少?

(2) 数组元素A[1][2]虚拟地址对应的外部页表项的物理地址是多少?若该外部页表项中存放的块号为00301H,则A[1][2]所在页面的页表项的物理地址是多少?

(3) 对数组A按行遍历和按列遍历,哪一种遍历方式的局部性更好?

6. (7分) 在Linux系统的shell中依次执行下列命令,所有命令执行完成后,给出各文件(包括目录文件)的i节点中count计数值

```
touch /tmp/f1  
mkdir /tmp/dir1  
mkdir /tmp/dir1/dir2  
ln /tmp/f1 /tmp/dir1/f2  
ln -s /tmp/f1 /tmp/dir1/f3  
ln /tmp/f1 /tmp/dir1/dir2/bar  
ln -s /tmp/dir1 /tmp/dir1/dir2/bar2
```

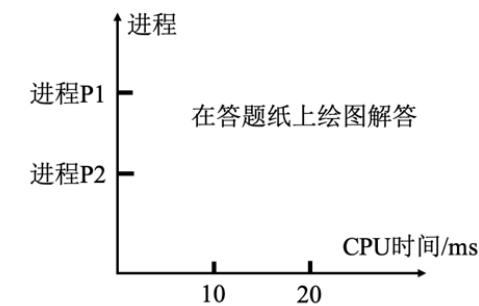
7. (11分) 系统采用二级反馈队列调度算法进行调度。就绪队列Q1采用时间片轮转调度算法,时间片为10ms;就绪队列Q2采用短进程优先调度算法;系统优先调度Q1队列中的进程,当Q1为空时系统才会调度Q2中的进程;新创建的进程首先进入Q1;Q1中的进程执行一个时间片后,若未结束,则转入Q2。若当前Q1,Q2为空,系统依次创建进程P1、P2后,即开始进行调度。P1、P2需要的CPU时间为30ms和20ms。

(1) 分析P1、P2的调度运行过程。

(2) 绘制P1、P2的调度运行过程图(如图所示)。

(3) 计算进程P1、P2在系统中的平均等待时间。

(4) 计算进程P1、P2在系统中的平均带权周转时间。



座位号:

答题卷

学号: 姓名: 成绩: _____

一、选择题 (每题 1 分, 共 25 分) 得分:

1.	2.	3.	4.	5.	6.	7.	8.	9.	10.
11.	12.	13.	14.	15.	16.	17.	18.	19.	20.
21.	22.	23.	24.	25.					

二、综合题 (共 75 分) 得分:

1(7).	2(11).	3(12).	4(14).
5(13).	6(7).	7(11).	

座位号:

答题卷

座位号:

答题卷

2023-2024-1 A 卷参考答案及评分标准

一、选择题：每小题 1 分，共 25 分

1. C	2. D	3. B	4. C	5. B	6. D	7. A	8. B	9. C	10. C
11. D	12. A	13. B	14. C	15. D	16. B	17. C	18. D	19. B	20. C
21. A	22. B	23. B	24. B	25. C					

评分标准：每小题答案正确得 1 分

二、综合题（共 75 分）

1. (7 分)

答：(1) (4 分)：因为文件不修改，在磁盘中连续存放，磁盘寻道时间更短，文件随机访问效率更高 (2 分)；

在 FCB 中加入的字段为〈起始块号，块数〉或者〈起始块号，结束块号〉。(2 分)

(2) (3 分)：为了快速找到文件，将所有的 FCB 集中存放好 (1 分)。

因为目录是存放在磁盘上的，将 FCB 集中存放，文件目录项中仅保留文件名和指向 FCB 的指针，课减少目录文件的磁盘块数量，检索文件目录时读取的磁盘块数量就减少了，从而加快目录检索速度 (2 分)。

2. (11 分)

(1) (3 分)：每个索引块中存放盘块号数量： $4KB/4B=1024$

文件最大长度：

$(12+1024+1024*1024+1024*1024*1024)*4KB=48KB+4MB+4GB+4TB$ 。

(2) (5 分)：文件索引节点的总个数为 $1M*4KB/64B=64M$ (2 分)

6000 的文件占 2 个磁盘块，512M 个磁盘块可存放文件的个数为 $512M/2=256M$ ，(2 分)

可表示文件的总个数受限于文件索引节点的总个数，因此只能存储 64M 个大小为 6000B 的视频文件。(1 分)

(3) (3 分)：Ext2 文件系统主要采用原地查找策略和预分配策略来解决文件存储空间碎片化的问题

3. (12 分) (1) (6 分)：错一个页面信息扣 1 分，全部正确得 6 分

页号	块号	修改位	访问位	引用时间	外存地址	存在位
0	66	0	1	1203	22	1
1	67	0	1	1178	23	1
2	87	0	1	1225	25	1
3	31	1	1	1020	26	1
4	—	—	—	—	—	0
5	1	0	1	1250	—	1

(2) (4分) 采用 LRU 算法, 将淘汰数据页 3 号页, 因为已经被修改, 需要先将该页写回磁盘; 修改页表, 然后将 31 号块分配给数据堆页, 修改页表相应内容, 返回访问数据堆页的指令, 重新执行。

(2分) 3号页面、4号页面页表项修改各 1 分

页号	块号	修改位	访问位	引用时间	外存地址	存在位
0	66	0	1	1203	22	1
1	67	0	1	1178	23	1
2	87	0	1	1225	25	1
3	31	1→—	1→—	1020→—	26	1→0
4	—→31	—→0	—→1	—→1270	—	0→1
5	1	0	1	1250	—	1

4. (14 分)

(1) (共 10 分)

```
// 信号量设置和初始值 2分
struct semaphore    pass_mutex=1,      east_count_mutex=1,
west_count_mutex=1;
int east_count=0, west_count=0;
```

从东向西: (4分)

```
P(east_count_mutex);
if(east_count==0)
P(pass_mutex);
east_count++;
V(east_count_mutex);
从东向西通过峡谷绳索
P(east_count_mutex);
east_count--;
if(east_count==0)
V(pass_mutex);
V(east_count_mutex);
```

从西向东: (4分)

```
P(west_count_mutex);
if(west_count==0)
P(pass_mutex);
west_count++;
V(west_count_mutex);
从西向东通过峡谷绳索
P(west_count_mutex);
west_count--;
if(west_count==0)
V(pass_mutex);
V(west_count_mutex);
```

(2) (共 4 分) 上述通行规则可能产生饥饿问题。当抢到绳索的这个方向的狒狒无限多或者非常多的情况下, 另一个方向的狒狒会出现无限等待, 即饥饿。(2分)

解决思路: 给出一种, 合理即可给分 (2分)。

思路 1: 控制一方连续通过的狒狒数量, 达到连续数量之后就需要让出绳索控制权, 重新争抢绳索。

思路 2: 当有一方狒狒等待的时候, 另一方没有上绳索的狒狒也不能上绳索, 必须等待绳索空之后重新争抢绳索。

5. (13 分)

(1) (6 分):

(2 分) 数组元素 A[1][2] 的虚拟地址:

$$\begin{aligned} \text{数组 A 起始虚拟地址} + (\text{行号} * 512 + \text{列号}) * 4 &= 1080\ 0000H + (1 * 512 + 2) * 4 \\ &= 1080\ 0000H + 2056_{(10)} = 1080\ 0000H + 0808H = 1080\ 0808H \end{aligned}$$

数组元素 A[1][2] 的虚拟地址 1080 0808H, 转换成二进制:

0001000010 0000000000 100000001000

外部页号: 042H (2 分)

外部页内地址: 0 (2 分)

(2) (5 分)

(2 分) 进程的外部页表起始地址为 00201000H, 外部页表项长度为 4 字节, 因此 A[1][2] 对应的外部页表项物理地址为: 00201000H + 4 * 042H = 0020 1108H

(3 分) A[1][2] 所在页面的页表项所在页表分页的页框号为 00301H, 每个页表项占 4 字节, 外部页内地址是 0, 因此页表项物理地址为: 00301H * 4K + 0 = 00301000H

(3) (2 分) 按行遍历的局部性更好。

6. (7 分)

每个文件 1 分

文件	f1	f2	f3	bar	bar2	dir1	dir2
Count	3	3	1	3	1	1	1

或者

文件	f1	f2	f3	bar	bar2	dir1	dir2
Count	3	3	1	3	1	3	2

7. (11 分)

(1) (4 分) 进程 P, P2 依次创建后进入队列 Q1, 根据时间片调度算法的规则, 进程 P, P2 将依次被分配 10ms 的 CPU 时间, 两个进程分别执行完一个时间片后都会被转入队列 Q2, 就绪队列 Q2 采用短进程优先调度算法, 此时 P 还需要 20ms 的 CPU 时间, P2 还需要 10ms 的 CPU 时间, 所以 P2 会被优先调度执行, 10ms 后进程 P2 执行完成, 之后 P 再调度执行, 再过 20ms 后 P1 也执行完成。运行图表述如下。

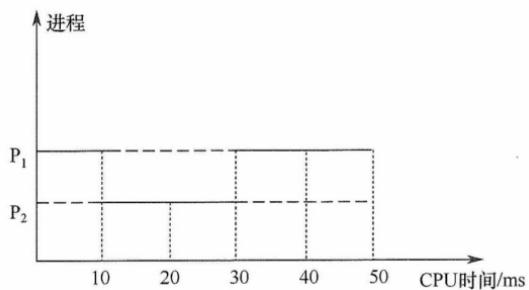
(2) (2 分) 绘图

(3) (2 分) 进程 P1、P2 的等待时间

分别为图中的虚横线部分,

平均等待时间 = (P1 等待时间 + P2 等待时间) / 2 = (20 + 10) / 2 = 15ms

(4) (3 分) 平均带权周转时间 = (P1 带权周转时间 + P2 带权周转时间) / 2



$$= [(50-0)/30 + (30-0)/20]/2$$

$$= (5/3+3/2)/2 = 19/12 \text{ 约 } 1.583\text{ms}$$