

新考点补充习题

注：若有后续勘误，将在下面这个文档中持续更新。

【腾讯文档】王道考前补充资料勘误表：<https://docs.qq.com/sheet/DRVJ5SGxzWk5PaEF3>

操作系统部分

注：1~20题我们曾公开分享过，有的同学可能已经做过了，可以跳过，也可以再复习一遍。

1. 用（ ）设计的操作系统结构清晰且便于调试。
 - A. 分层式构架
 - B. 模块化构架
 - C. 微内核构架
 - D. 宏内核构架
2. 下列关于层次式结构操作系统的说法中，（ ）是错误的。
 - A. 各层之间只能是单向依赖或单向调用
 - B. 容易实现在系统中增加或替换一层而不影响其他层
 - C. 具有非常灵活的依赖关系
 - D. 系统效率较低
3. 相对于微内核系统，（ ）不属于大内核操作系统的缺点。
 - A. 占用内存空间大
 - B. 缺乏可扩展性而不方便移植
 - C. 内核切换太慢
 - D. 可靠性较低
4. 下列说法中，（ ）不适于描述微内核操作系统。
 - A. 内核足够小
 - B. 功能分层设计
 - C. 基于C/S模式
 - D. 策略与机制分离
5. 对于以下四种服务，在采用微内核结构的操作系统中，（ ）不宜放在微内核中。
 - I. 进程间通信机制
 - II. 低级I/O
 - III. 低级进程管理和调度
 - IV. 中断和陷入处理
 - V. 文件系统服务
 - A. I、II和III
 - B. II和V
 - C. 仅V
 - D. IV和V
6. 下列关于操作系统结构说法中，正确的是（ ）。
 - I. 当前广泛使用的Windows XP操作系统，采用的是分层式OS 结构
 - II. 模块化的OS 结构设计的基本原则是：每一层都仅使用其底层所提供的功能和服务，这样使系统的调试和验证都变得容易
 - III. 由于微内核结构能有效支持多处理机运行，故非常合适于分布式系统环境

IV. 采用微内核结构设计和实现操作系统具有诸多好处, 如添加系统服务时, 不必修改内核、使系统更高效。

A. I和II B. I和III C. III D. III和IV

7. 计算机操作系统引导, 描述不正确的是 ()。

- A. 计算机的引导程序驻留在ROM中, 开机后便自动执行
- B. 引导程序先做关键部位的自检, 并识别已连接的外设
- C. 引导程序会将硬盘中存储的操作系统全部加载到内存
- D. 若计算机中安装了双系统, 引导程序会与用户交互加载有关系统

8. 计算机操作系统的引导程序位于 ()。

A. 主板BIOS B. 片外Cache C. 主存ROM区 D. 硬盘

9. 计算机的启动过程是 ()。①CPU加电, CS:IP指向FFFF0H; ②进行操作系统引导; ③执行JMP指令跳转到BIOS; ④登记BIOS中断例程入口地址; ⑤硬件自检。

A. ①②③④⑤ B. ①③⑤④② C. ①③④⑤② D. ①⑤③④②

10. 检查分区表是否正确, 及确定哪个分区为活动分区, 并在程序结束时把该分区的启动程序 (操作系统引导扇区) 调入内存加以执行, 这是 () 的任务。

A. MBR B. 引导程序 C. 操作系统 D. BIOS

11. 下列关于虚拟机的说法中, 正确的是 ()。

- I. 虚拟机可以用软件实现 II. 虚拟机可以用硬件实现
 - III. 多台虚拟机可同时运行在同一物理机器, 它实现了真正的并行
- A. I和II B. I和III C. 仅I D. I、II和III

12. 下列关于VMware workstation虚拟机的说法中, 错误的是 ()。

- A. 真实硬件不会直接执行虚拟机中的敏感指令
- B. 虚拟机中只能安装一种操作系统
- C. 虚拟机是运行在计算机的一个应用程序
- D. 虚拟机文件封装在一个文件夹中, 并存储在数据存储器

13. 下面关于用户级线程和内核级线程的描述中, 错误的是 ()。

- A. 采用轮转调度算法, 进程中设置内核线程和用户线程的效果完全不同
- B. 跨进程的用户级线程调度也不需要内核参与, 控制简单
- C. 用户级线程可以在任何操作系统中运行
- D. 若系统中只有用户级线程, 则处理机的调度对象是进程

14. 对内核级线程相对于用户级线程的优点的描述中, 错误的是 ()

- A. 同一进程内的线程切换, 系统开销小
- B. 当内核线程阻塞时, CPU将会调度同一进程中的其他内核线程执行

- C. 内核级线程的程序实体可以在内核态运行
- D. 对多处理器系统，核心可以同时调度同一进程的多个线程并行运行
15. 下面对用户级线程相比内核级线程的优点描述中，错误的是（ ）
- A. 一个线程阻塞不影响另一个线程的运行
- B. 线程的调度不需要内核直接参与，控制简单
- C. 线程切换代价小
- D. 允许每个进程定制自己的调度算法，线程管理比较灵活
16. 进程（线程）调度的时机有（ ）。
- I. 运行的进程（线程）运行完毕 II. 运行的进程（线程）所需资源未准备好
- III. 运行的进程（线程）的时间片用完 IV. 运行的进程（线程）自我阻塞
- V. 运行的进程（线程）出现错误
- A. II、III、IV和V B. I和III C. II、IV和V D. 全部都是
17. 下列选项中，（ ）不是Linux实现虚拟文件系统VFS所定义的对象类型。
- A. 超级块（superblock）对象 B. 目录项（inode）对象
- C. 文件（file）对象 D. 数据（data）对象
18. 硬盘的操作系统引导扇区产生在（ ）。
- A. 对硬盘进行分区时 B. 对硬盘进行低级格式化时
- C. 硬盘出厂时自带 D. 对硬盘进行高级格式化时
19. 下列关于固态硬盘（SSD）的说法中，错误的是（ ）。
- A. 基于闪存的存储技术 B. 随机读写性能明显高于磁盘
- C. 随机写比较慢 D. 不易磨损
20. 下列关于固态硬盘的说法中，正确的是（ ）。
- A. 固态硬盘的写速度比较慢，性能甚至弱于常规硬盘
- B. 相比常规硬盘，固态硬盘优势主要体现在连续存取的速度
- C. 静态磨损均衡算法比动态磨损均衡算法的表现更优秀
- D. 写入时，每次选择使用长期存放数据而很少擦写的存储块
21. 关于操作系统外核（exokernel），错误的是（ ）
- A. 外核可以给用户进程分配未经抽象的硬件资源
- B. 用户进程通过调用“库”请求操作系统外核的服务
- C. 外核负责完成进程调度
- D. 外核可以减少虚拟硬件资源的“映射”开销，提升系统效率

22. 下列选项中, () 不属于模块化操作系统的特点

- A、很多模块化的操作系统, 可以支持动态加载新模块到内核, 适应性强
- B、内核中的某个功能模块出错不会导致整个系统崩溃, 可靠性高
- C、内核中的各个模块, 可以相互调用, 无需通过消息传递进行通信, 效率高
- D、各模块间相互依赖, 相比于分层式操作系统, 模块化操作系统更难以调试和验证

23. 虚拟机的实现, 离不开虚拟机管理程序 (VMM), 下列说法中正确的是 ()

- ①第一类虚拟机管理程序直接运行在硬件之上, 通常效率高于第二类虚拟机管理程序
- ②由于虚拟机管理程序的上层需要支持操作系统的运行、应用程序的运行, 因此实现虚拟机管理程序的代码量通常大于实现一个完整的操作系统
- ③虚拟机管理程序可将一台物理机器虚拟化为多台虚拟机器
- ④为了支持客户操作系统的运行, 第二类虚拟机管理程序需要完全运行在最高特权级

- A、①②③
- B、①③
- C、①②③④
- D、①③④

答案及解析

1. A

层次化结构简化了系统的设计和实现, 每层只能调用紧邻它的低层的功能和服务; 便于系统的调试和验证, 在调试某层时发现错误, 那么错误应在这层上, 这是因为其低层都已调试好了。

2. C

单向依赖是层次式OS的特点。层次式OS中增加或替换一个层次中的模块或整个层次时, 只要不改变相应层次间的接口, 就不会影响其他层, 因而易于扩充和维护。层次定义好后, 相当于各层次之间的依赖关系也就固定了, 因此往往显得不够灵活, C错误。每执行一个功能, 通常要自上而下穿越多个层, 增加了额外的开销, 导致系统效率降低。

3. C

微内核和宏内核作为两种对立的结构, 他们的优缺点也是相对立的。微内核OS的主要缺点是性能问题, 因为需要频繁地在核心态和用户态之间进行切换, 因而切换开销偏大。

4. B

功能分层设计是分层式OS的特点。通常可以从四个方面来描述微内核OS: ①内核足够小; ②基于客户/服务器模式; ③应用“机制与策略分离”原理; ④采用面向对象技术。

5. C

进程（线程）之间的通信功能是微内核最频繁使用的功能，因此几乎所有微内核OS都将其放入微内核中。低级I/O和硬件紧密相关，因此应放入微内核中。低级进程管理和调度属于调度功能的机制部分，应将它放入微内核中。微内核OS将与硬件紧密相关的一小部分放入微内核中处理，此时微内核的主要功能是捕获所发生的中断和陷入事件，并进行中断响应处理，在识别中断或陷入的事件后，再发送给相关的服务器来处理，故中断和陷入处理也应放入微内核中。而文件系统服务是放在微内核外的文件服务器中实现的，故仅V不宜放在微内核中。

6. C

Windows是宏内核操作系统，I错误。II描述的是层次化构架的原则。在微内核构架中，客户和服务端之间、服务器和服务端之间的通信采用消息传递机制，就使得微内核系统能很好地支持分布式系统，III正确。添加系统服务时不必修改内核，这使得微内核构架的可扩展性和灵活性更强；微内核构架的主要问题是性能问题，“使系统更高效”显然错误。

7. C

操作系统开机时，需要依次执行：①ROM引导程序、②磁盘引导程序、③分区引导程序、④操作系统初始化程序。其中，操作系统初始化程序会把操作系统全部加载到内存

8. D

操作系统的引导程序位于磁盘活动分区的引导扇区。引导程序可理解为分成两种：一种是位于ROM中的自举程序（BIOS的组成部分），用于启动具体的设备；另一种是位于装有操作系统硬盘的活动分区引导扇区中的引导程序（称为启动管理器），用于引导操作系统。

9. C

CPU激活后，会从最顶端的地址FFFF0H获得第一条指令来执行，这个地址仅仅只有16字节，放不下一段程序，所以是一条JMP指令，跳到更低地址去执行BIOS程序。BIOS程序在内存最开始的空间构建中断向量表和相应服务程序，在后续POST过程中要用到中断调用等功能。然后进行通电自检POST（Power-on Self Test）以检测硬件是否有故障。完成POST后，BIOS需要在硬盘、光驱、软驱等存储设备搜寻操作系统内核的位置以启动操作系统。

10. A

BIOS将控制权交给排在首位的启动设备后，CPU将该设备主引导扇区的内容（主引导记录MBR）加载到内存中，然后由MBR检查分区表，查找活动分区，并将该分区的引导扇区的内容（分区引导记录PBR）加载到内存加以执行。

11. A

软件能实现的功能也能由硬件实现，因为虚拟机软件能实现的功能也能由硬件实现，软件和硬件的分界面是系统结构设计者的任务，I和II正确。实现真正并行的是多核处理器，多台虚拟机同时运行在同一物理机器上，类似于多个程序运行于同一系统中。

12. B

VMware workstation虚拟机属于第二类虚拟机管理程序，如果真实硬件直接执行虚拟机中的敏感指令，那么若该指令非法可能会导致宿主操作系统的崩溃，而这是不可能的，实际上是由第二类虚拟机管理程序模拟真实硬件环境。虚拟机看起来和真实物理计算机没什么两样，因此当然可以安装多个操作系统。VMware workstation就是一个安装在计算机上的程序，在创建虚拟机时，会为该虚拟机创建一组文件，这些虚拟机文件都存储在主机的磁盘上。

13. B

采用轮转调度算法时，如果进程中分别设置100个用户级线程和100个内核级线程，则前者的执行时间是后者的100倍，A正确。在用户级线程中，系统感知不到线程的存在，调度的对象是进程，因此，跨进程的用户级线程调度需要内核参与，B错误，D正确。用户级线程不需要内核的支持，与系统平台无关，对线程管理的代码是属于用户程序的一部分，C正确。

14. A

在内核级线程中，同一进程中的线程切换，需要从用户态转到核心态进行，系统开销较大，A错误。CPU调度是在内核进行的，在内核级线程中，调度是在线程一级进行，因此内核可以同时调度同一进程的多个线程在多CPU上并行运行（用户级线程则不行），B正确、D正确。当进程中的内核级线程运行在内核态时，则说明该进程也运行在内核态，C正确。

15. A

进程中的某个用户级线程阻塞，则整个进程也会被阻塞，即进程中的其他用户级线程也被阻塞，A错误。用户级线程的调度是在用户空间进行的，节省了模式切换的开销，不同进程可根据自身需要，对自己的线程选择不同的调度算法，因此B、C和D都正确。

16. D

进程（线程）调度的时机有运行的进程（线程）运行完毕、运行的进程（线程）自我阻塞、运行的进程（线程）的时间片用完、运行的进程（线程）所需资源没有准备好、运行的进程（线程）出现错误；以上都是在CPU为不可抢占方式下引起进程（线程）调度。在CPU方式是可抢占方式时，就绪队列中的某个进程（线程）的优先级高于当前运行进程（线程）的优先级时，也会发生进程（线程）调度。故I、II、III、IV和V都正确。

17. D

为了实现虚拟文件系统（VFS），Linux主要抽象了四种对象类型：超级块对象、索引结点对象、目录项对象和文件对象。D错误。

18. D

操作系统的引导程序位于磁盘活动分区的引导扇区，因此必然产生在分区之后。分区是将磁盘分为由一个或多个柱面组成的分区（即C盘、D盘等形式），每个分区的起始扇区和大小都记录在磁盘主引导记录的分区表中。而高级格式化（创建文件系统），操作系统将初始的文件系统数据结构存储到磁盘上，文件系统在磁盘的布局在第四章文件系统中介绍过。

19. D

固态硬盘基于闪存技术，没有机械部件，随机读写不需要机械操作，因此速度明显高于磁盘，A和B正确。选项C已在考点讲解中解释过。SSD的缺点是容易磨损，D错误。

20. C

SSD的写速度慢于读速度，但不至于比常规机械硬盘差，A错误。SSD基于闪存技术，没有机械部件，随机存取速度很快，传统机械硬盘因为需要寻道和找扇区的时间，所以随机存取速度慢；传统机械硬盘转速很快，连续存取比随机存取快得多，因此SSD的优势主要体现在随机存取的速度，B错误。静态磨损算法在没有写入数据时，SSD监测并自动进行数据分配，因此表现更为优秀，C正确。因闪存的擦除速度较慢，若每次都选择写入存放有数据的块，会极大降低写入速度，选项D混淆了静态磨损均衡，静态磨损均衡是指在写入数据时，SSD监测并自动进行数据分配，从而使得各块的擦写更加均衡，并不是说写入时每次选择存放老数据的块。

21.C

在拥有外核的操作系统中，外核只负责硬件资源的分配、回收、保护等，进程管理相关的工作仍然由内核负责。

22.B

模块化操作系统，各功能模块都在内核中，且模块间相互调用相互依赖，任何一个模块出错，都有可能整个内核的崩溃。

该选项的设置属于“移花接木”，正确的说法是：在微内核操作系统中，内核外的某个功能模块出错不会导致整个系统崩溃，可靠性高。

各种操作系统结构的总结如下：

操作系统结构			
	特性、思想	优点	缺点
分层结构	内核分多层，每层可单向调用更低一层提供的接口	<ul style="list-style-type: none"> 1. 便于调试和验证，自底向上逐层调试验证 2. 易扩充和易维护，各层之间调用接口清晰固定 	<ul style="list-style-type: none"> 1. 仅可调用相邻低层，难以合理定义各层的边界 2. 效率低，不可跨层调用，系统调用执行时间长
模块化	<p>将内核划分为多个模块，各模块之间相互协作。</p> <p>内核 = 主模块 + 可加载内核模块</p> <p>主模块：只负责核心功能，如进程调度、内存管理</p> <p>可加载内核模块：可以动态加载新模块到内核，而无需重新编译整个内核</p>	<ul style="list-style-type: none"> 1. 模块间逻辑清晰易于维护，确定模块间接口后即可多模块同时开发 2. 支持动态加载新的内核模块（如：安装设备驱动程序、安装新的文件系统模块到内核），增强OS适应性 3. 任何模块都可以直接调用其他模块，无需采用消息传递进行通信，效率高 	<ul style="list-style-type: none"> 1. 模块间的接口定义未必合理、实用 2. 模块间相互依赖，更难调试和验证
宏内核（大内核）	所有的系统功能都放在内核里（大内核结构的OS通常也采用了“模块化”的设计思想）	<ul style="list-style-type: none"> 1. 性能高，内核内部各种功能都可以直接相互调用 	<ul style="list-style-type: none"> 1. 内核庞大功能复杂，难以维护 2. 大内核中某个功能模块出错，就可能整个系统崩溃
微内核	只把中断、原语、进程通信等最核心的功能放入内核。进程管理、文件管理、设备管理等功能以用户进程的形式运行在用户态	<ul style="list-style-type: none"> 1. 内核小功能少、易于维护，内核可靠性高 2. 内核外的某个功能模块出错不会导致整个系统崩溃 	<ul style="list-style-type: none"> 1. 性能低，需要频繁的切换用户态/核心态。用户态下的各功能模块不可以直接相互调用，只能通过内核的“消息传递”来间接通信 2. 用户态下的各功能模块不可以直接相互调用，只能通过内核的“消息传递”来间接通信
外核（exokernel）	内核负责进程调度、进程通信等功能，外核负责为用户进程分配未经抽象的硬件资源，且由外核负责保证资源使用安全	<ul style="list-style-type: none"> 1. 外核可直接给用户进程分配“不虚拟、不抽象”的硬件资源，使用户进程可以更灵活的使用硬件资源 2. 减少了虚拟硬件资源的“映射层”，提升效率 	<ul style="list-style-type: none"> 1. 降低了系统的一致性 2. 使系统变得更复杂

23.B

第一类VMM直接运行在硬件上，直接和硬件打交道；第二类VMM运行在宿主操作系统之上，不能直接和硬件打交道。因此通常第一类VMM效率更高，①正确。虚拟机管理程序的功能没有操作系统的功能复杂，因此开发虚拟机管理程序的代码量小于开发一个完整的操作系统，②错误。③是基本概念，正确。第一类VMM运行在最高特权级（内核态），而第二类VMM和普通应用程序的地位相同，通常运行在较低特权级（用户态），④错误。

关于两类VMM的对比总结如下：

两类虚拟机管理程序（VMM）的对比

	第一类VMM	第二类VMM
对物理资源的控制权	直接运行在硬件之上，能直接控制和分配物理资源	运行在Host OS之上，依赖于Host OS为其分配物理资源
资源分配方式	在安装Guest OS时，VMM要在原本的硬盘上自行分配存储空间，类似于“外核”的分配方式，分配未经抽象的物理硬件	GuestOS 拥有自己的虚拟磁盘，该盘实际上是Host OS 文件系统中的一个文件。GuestOS分配到的内存是虚拟内存
性能	性能更好	性能更差，需要HostOS作为“中介”
可支持的虚拟机数量	更多，不需要和 Host OS 竞争资源，相同的硬件资源可以支持更多的虚拟机	更少，Host OS 本身需要使用物理资源，Host OS 上运行的其他进程也需要物理资源
虚拟机的可迁移性	更差	更好，只需导出虚拟机镜像文件即可迁移到另一台 HostOS 上，商业化应用更广泛
运行模式	第一类VMM运行在最高特权级（Ring 0），可以执行最高特权的指令。	第二类VMM部分运行在用户态、部分运行在内核态。GuestOS 发出的系统调用会被 VMM 截获，并转化为 VMM 对 HostOS 的系统调用