

一、名词解释题

- 1、**并发与并行**：并发指多个任务在多个处理机上正在同时运行，并行指多个任务在单处理机下分时运行。
- 2、**临界资源与临界区**：临界资源指某段时间内一次仅允许一个进程使用的资源，临界区指临界资源的那段程序代码。
- 3、**系统调用**：在操作系统核心设置的一组用于实现各种系统功能的子程序（过程）。
- 4、**进程互斥**：指在多道环境中，每次只允许一个进程对临界资源访问。
- 5、**中断屏蔽**：指中断请求产生后，系统用软件的方式，有选择地封锁部分中断而允许其他部分的中断仍然能够得到响应。
- 6、**目录**：目录是保存目录结构的信息文件，在目录文件中保存着该目录所包含的目录或文件记录，每个记录包括目录或文件的名称、大小、存储位置、存取权限及其他相关数据项。
- 7、**原语**：由若干条指令所组成，来实现某个特定的操作，通过一段不可分割的，或者不可中断的程序实现其功能。
- 8、**快表**：存在于地址变换机构中的一个由高速寄存器组成的小容量的联想寄存器，构成的一张表。
- 9、**设备无关性**：程序可以通过一组统一的操作过程来操作设备。这种操作接口与具体的设备无关。
- 10、**文件系统**：一个负责存取和管理外部存储器上的文件信息的机制。
- 11、**文件控制块**：文件控制块是操作系统为管理文件而设置的数据结构，存放为了管理文件所需的所有有关信息，文件控制块是文件存在的标志。文件控制块一般包括：（文件名、文件类型、物理地址、文件大小、最近访问日期、最近修改日期、文件主标识、访问权限）
- 12、**虚拟存储器**：虚拟存储技术是在主存和辅存之间，增加部分软件及必要的硬件支持，使主、辅之间的信息交换、程序重定位、地址转换都能自动地进行。从而使主、辅存形成一个有机的整体，这种存储器的概念称为虚拟存储器。
- 13、**死锁**：两个程序相互等待一个永远不可能发生的条件出现，这种僵持叫做死锁。
- 14、**页表**：页式存储管理中使用的数据结构，主要用于逻辑地址到物理地址的映射。
- 15、**进程**：进程是进程实体的运行过程，是系统进行资源分配和调度的一个独立单位。这就要求进程调度程序按一定的策略，动态地把处理机分配给处于就绪队列中的某一个进程，以使之执行。
- 16、**系统调用**：操作系统与程序之间的接口。
- 17、**进程控制块 PCB**：进程控制块是一个与动态过程相联系的数据结构，记载了进程的外部特性（名称、状态等）以及与其他进程的联系（通信关系），还记录了进程所拥有的各种资源，进程控制块是进程存在的标志。
- 18、**缓冲区**：为了解决外部设备和内存或外部设备和 CPU 之间的数据传输速度不匹配的问题，在系统中引入缓冲区来暂存数据。
- 19、**文件目录**：目录是文件系统层次结构的一个非终结节点，一个目录通常包括许多目录项，每个目录项可能是一个文件或目录。
- 20、**特权指令**：指具有特殊权限的指令，主要用于系统资源的分配和管理。包括改变系统工作方式，检测用户的访问权限，修改虚存的段表和页表等。
- 21、**多道程序设计**：是在计算机内存中同时存放几道相互独立的程序，使它们在管理程序控制之下，相互穿插的运行。两个或两个以上程序在计算机系统中同处于开始到结束之间的状态。间断性、共享性、制约性

二.判断题

1. 用 P、V 操作可以解决一切互斥与同步问题。(T)
2. 同一进程或不同进程内的线程都可以并发执行。(T)
3. 采用多道程序设计技术的计算机系统,极大地提高了计算机系统的系统效率,但可能使每个作业的执行时间延长。(T)
4. 作业调度的先来先服务算法,按照作业到达的先后次序调度作业,排队等待时间最长的作业被优先调度。(T)
5. 采用 SPOOLing 技术实现的共享设备,在同一时刻可以让多个进程使用它进行 I/O。(F)
6. 设备独立性(或无关性)是指能独立实现设备共享的一种特性。(F)
7. 临界区的执行不能被中断。(F)
8. 资源顺序分配法破坏了死锁发生的循环等待必要条件。(T)
9. 对磁盘进行磁头调度的目的是为了缩短寻道时间。(T)
10. 采用页式存储管理时,重定位的工作是由用户完成的。(F)
11. 与设备相关的中断处理过程由设备驱动程序完成。(T)
12. 由于 P、V 操作描述同步、互斥等问题的能力不足,所以有必要引入其它的通讯原语或机制,如 send, receive 或 Monitor 等。(X)
13. 信号量是只允许由 P/V 操作进行访问和修改的数据结构。(✓)
14. 在请求页式存储管理中,页面淘汰所花费的时间不属于系统开销。(X)
15. 预防死锁就是破坏死锁存在的某个必要条件。(✓)
16. 磁盘是一类典型的字符设备。(X)
17. 银行家算法是一种避免死锁的算法。(T)
18. 分时系统中,时间片越小越好。(F)
19. 并发是指若干事件在同一时间间隔内发生。(T)
20. 在多道程序系统中,道数越多越好。(F)
21. 打印机是一类典型的字符设备。(T)
22. 一个进程可以涉及一个或若干个程序的执行;反之,同一个程序只可以对应一个进程。(X)
23. 信号量是只允许由 P/V 操作进行访问和修改的数据结构。(✓)
24. 并发是指多个任务在多个处理机上正在同时运行,在微观上看,这些任务是在各自的物理处理机上分别运行。(X)
25. 进程的同步与互斥可以发生在一个进程之中。(X)
26. 中断方式的数据传送是在中断处理时由 CPU 控制完成的;DMA 方式则不经过 CPU,而是在 DMA 控制器的控制下完成的。(✓)
27. 动态重定位便于程序浮动,其实现时采用的硬件机构是重定位寄存器和加法器。(✓)
28. 并发进程的执行结果只取决于进程本身,不受外界影响。(F)
29. 任何一个进程在申请新资源前总是先归还已得到的资源,则系统不会死锁。(T)
30. P、V 操作不仅可用来实现进程的同步与互斥,而且可以防止系统死锁。(F)
31. 银行家算法是在保证至少有一个进程能得到所需的全部资源的前提下进行资源分配的。(T)
32. 如果不能控制并发进程执行的相对速度,则它们在共享资源时一定会出现与时间有关的错误。(F)

三.简答题

1.何谓缓冲区？为什么要引入缓冲？

答：缓冲区是使用专用硬件缓冲器或在内存中划出一个区域用来暂时存放输入/输出数据的地方。引入缓冲是为了匹配外设和 CPU 之间的处理速度，减少中断次数和 CPU 的中断处理时间。

2. 什么是死锁？产生死锁的必要条件是什么？

答：两个以上的进程相互等待一个永远不可能发生的条件，这种僵持局面称 死锁。死锁产生的必要条件：互斥条件、不剥夺条件、请求和保持条件、循环等待条件。

3.DMA 方式与中断方式的不同点

答：（1）中断方式在每个数据传送完后中断 CPU，而 DMA 方式则是在所要求传送的一批数据全部传送结束时中断 CPU。（2）中断方式的数据传送是在中断处理时由 CPU 控制完成，而 DMA 方式则是在 DMA 控制器下完成的。

4.什么是重定位？如何实现程序运行时的动态重定位？

答：所谓重定位就是把作业的地址空间中的相对地址转换成内存空间的物理地址的调整过程。在程序实际运行前，由操作系统把程序在内存的开始地址放入重定位寄存器。在程序运行期间，凡遇到访问内存的操作，就由硬件机制自动把用户的程序的相对地址加上重定位寄存器的内容，相加之和就是实际访问内存的有效地址。

5.进程的含义是什么？如何描述和构造进程？

答：进程是程序的一次执行，进程由“进程控制块+程序+数据”构成，用进程控制块描述。

6.什么是开中断，什么是关中断？

答：尽管产生了中断源和发出了中断请求，但 CPU 内部处理机状态字的中断允许位已被清除，从而不允许 CPU 响应中断，这种情况被称为关中断。

CPU 禁止中断后只有等到处理机状态字的中断允许位被重新设置后才能接收中断，处理机的状态位的设置称为开中断。

7.分页存储管理中有哪几种常用的页面转换算法？

答：先进先出算法(FIFO)、最近最少使用淘汰算法(LRU)、最近不经常使用淘汰算法(LFU)、最优算法(OPT)。

8.如果普通用户可以自行修改页表，会产生什么问题？

答：页表用于完成地址映射，如果用户可以修改页表，那么该用户就可以访问任何地址从而产生安全问题。

9.进程与线程之间有何区别？

答：进程是操作系统中的并发单元，也是能分得资源的最小单位。线程是在进程内部活动的并发单元，它只是进程行为的一条独立的执行路线，它所能使用的资源仅限于它所在的进程范围之内，惟一能通过线程获得的资源就是使用处理机的时间片，有时也把线程叫做轻量级线程。

10. 简述并比较 SCAN（扫描）磁盘调度算法与最短寻道时间优先算法

答：最短寻道优先算法选择访问磁道与当前磁头所在磁道距离最近的进程，容易产生饥饿现象。SCAN 优先考虑磁头的移动方向（按照一个方向移动）。

11.信号量的物理意义是什么？

答：信号量为正时，表示系统中某类资源的数量，为负时，表示等待进程的个数。

12. 为什么说链接式文件不适合随机存取？

答：访问链式结构文件中的任何一块都要求从文件的第一块开始向后顺序访问，直到找到要访问的那一块，因此，链式文件不适合随机存取。

13. 分页存储管理有效解决了什么问题？

答：分页存储管理有效解决了程序离散存放和存储器碎片问题，提高了存储器的利用率。

14. 实时系统和分时系统各有什么特点？有什么本质的区别

答：（1）实时系统通常是一个专用系统，它的特点是响应时间快，快的程度依赖于实时系统的种类，如果是实时控制系统，则响应时间依赖于实时控制对象的需求，根据需求及时响应，如果是实时信息管理系统，其响应时间与分时系统要求相似，一般不超过 3 秒，实时系统对安全性要求较高，系统的可靠安全是实时系统的保障。（2）分时系统亦称交互式系统，其特点是对用户响应及时，当多个用户使用计算机时，都有独占的感觉。（3）实时系统对响应时间的要求比分时系统更高，一般要求响应时间为秒级、毫秒级甚至微秒级。与分时系统相比，实时系统没有那么强的交互会话功能，通常不允许用户通过实时终端设备去编写新的程序或修改已有程序，实时终端设备只是作为执行装置或询问装置，属专用系统。

15. 简述设备管理的主要功能

答：（1）提供设备管理程序和进程管理系统的接口，当进程申请设备资源时，该接口将进程的请求转发给设备管理程序。（2）进行设备分配，按照设备类型和相应的分配算法，把设备和其他相关的硬件分配给请求该设备的进程，并把未分配到所请求设备的进程放入等待队列。（3）实现设备和设备、设备和 CPU 之间的并行操作，针对相应的硬件支持，采用不同的输入/输出控制方式。（4）进行缓冲区的管理，设备管理程序负责进行缓冲区的分配、释放及有关的管理工作。

16. 简述段页式存储的基本原理

答：段页式系统的基本原理是分段和分页原理相结合。即先将用户程序分为若干个段，再把每个段分为若干个页，并为每个段赋予一个段名，在段页式系统中，为了实现从逻辑地址到物理地址的转换，系统中需同时配置段表和页表，段表的内容还包括页表的起始地址和长度。

17. 什么是文件的物理结构？常见的文件物理组织有几种？

答：（1）文件的物理结构是指文件记录在文件管理系统内部采用的、与物理存储介质的特性相适应的方式，是为系统所使用的。（2）顺序文件结构、随机文件结构、串联文件。

18. 操作系统在进程管理方面的五项主要活动是什么？

答：进程控制、进程同步、进程调度、进程通信、死锁

19. 操作系统在存储管理方面有哪三项主要活动？

答：

20. 操作系统在外存管理方面有哪三项主要活动？

答：外存资源管理、外存空间划分、外存空间分配。

21. 操作系统在文件管理方面有哪五项主要活动

答：目录管理、文件存储空间管理、文件共享、文件保护、数据一致性控制

什么是 I/O 控制？它的主要任务是什么？

答：输入输出处理过程分为两个阶段，即：

用户 I/O 请求→分配设备→启动设备进行 I/O 操作

响应设备中断→设备中断处理

从处理 I/O 请求开始到设备中断处理结束，这一总过程称为 I/O 控制。

用于设备分配的数据结构有哪些？它们之间的关系是什么？

答：整个系统有一张 SDT，每个设备有一张 DCT，每个控制器有一张 COCT，每个通道有一张 CHCT。

I/O 控制可用哪几种方式实现？各有什么优缺点？

答：I/O 控制过程可用三种方式实现：作为请求 I/O 操作的进程实现；作为当前进程的一部

分实现：由专门的系统进程（I/O 进程）完成。

第一种方式请求对应 I/O 操作的进程能很快占据处理机，但要求系统和 I/O 操作的进程应具有良好的实时性。第二种方式不要求系统具有高的实时性，但 I/O 控制过程要由当前进程负责。第三种方式增加了一个额外的进程开销，但用户进程不用处理 I/O 控制过程。

4、设备驱动程序是什么？为什么要有设备驱动程序？用户进程怎样使用驱动程序？

答：设备驱动程序是驱动外部物理设备和相应 DMA 控制器或 I/O 控制器等器件，使之可以直接和内存进行 I/O 操作的子程序的集合。它们负责设置相应设备有关寄存器的值，启动设备进行 I/O 操作，指定操作的类型和数据流向等。

设备驱动程序屏蔽了直接对硬件操作的细节，为编程者提供操纵设备的便利接口。

用户进程通过调用设备驱动程序提供的接口来使用设备驱动程序。

12、**进程实体**有哪几部分组成？PCB、程序段和相关的数据段

13、**操作系统的分类**：批处理操作系统、分时操作系统和实时操作系统，批处理操作系统、分时操作系统和实时操作系统

14 什么是管态和目态？用户程序如何进入管态执行

大多数计算机系统将 CPU 执行状态分为管态和目态。管态又称为特权状态、系统态或核心态。通常，操作系统在管态下运行。目态又叫做常态或用户态，用户程序只能在目态下运行，如果用户程序在目态下执行特权指令，硬件将发生中断，由操作系统获得控制，特权指令执行被禁止，这样可以防止用户程序有意或无意的破坏系统。从目态转换为管态的唯一途径是中断。

15、试述页式存储管理的基本原理① 内存划分。② 逻辑地址空间划分。

③ 页面大小。④ 内存分配。

(2) 试述页式存储管理的实现方法① 建立页表。② 建立空闲页面表。

③ 硬件支持。④ 地址映射过程。

3.银行家算法

假设有 A、B、C、D 四类资源，在银行家算法中，若出现如下资源分配情况：

Process	Allocation	Need	Available
P0	0032	0012	1623
P1	1000	1750	
P2	1354	2356	
P3	0332	0652	
P4	0014	0656	

当前状态是否是安全的？若是，给出一个安全序列。

如果进程 P2 提出安全请求 Request[2]=(1,2,2,2)，系统能否将资源分配给它？说明原因。

(1) 当前状态是安全状态。

令 $Work = Available = (1, 6, 2, 3)$ ，运行安全性检测算法：

1) $Finish[0]=false$ 并且 $Need[0]=(0, 0, 1, 2) < Work$ ，则 $Work = Work + Allocation[0] = (1, 6, 2, 3) + (0, 0, 3, 2) = (1, 6, 5, 5)$ ； $Finish[0] = true$ ；

2) $Finish[3]=false$ 并且 $Need[3] = (0, 6, 5, 2) < Work$ ，则 $Work = Work + Allocation[3] = (1, 6, 5, 5) + (0, 3, 3, 2) = (1, 9, 8, 7)$ ； $Finish[3] = true$ ；

3) $Finish[4]=false$ 并且 $Need[4] = (0, 6, 5, 6) < Work$ ，则 $Work = Work + Allocation[4] = (1, 9, 8, 7) + (0, 0, 1, 4) = (1, 9, 9, 11)$ ； $Finish[4] = true$ ；

4) $Finish[1]=false$ 并且 $Need[1] = (1, 7, 5, 0) < Work$ ，则 $Work = Work + Allocation[1] = (1, 9, 9, 11) + (1, 0, 0, 0) = (2, 9, 9, 11)$ ； $Finish[1] = true$ ；

5) $Finish[2]=false$ 并且 $Need[2] = (2, 3, 5, 6) < Work$ ，则 $Work = Work + Allocation[2] = (2, 9, 9, 11) + (1, 3, 5, 4) = (3, 12, 14, 15)$ ； $Finish[2] = true$ ；

因此，可以找到一个安全进程序列 $\langle p0, p3, p4, p1, p2 \rangle$ ，它使对于所有 $0 \leq i \leq 4$ ， $Finish[i]=true$ ，因而系统当前处于安全状态。

(2) 运行银行家算法，由于 $Request[2] = (1, 2, 2, 2) \leq Need[2] = (2, 3, 5, 6)$ ，因而请求合法。进一步， $Request[2] = (1, 2, 2, 2) \leq Available = (1, 6, 2, 3)$ ，故该请求是可以满足的。假设将资源分配给 $p2$ ，则系统状态变为： $Available = (0, 4, 0, 1)$ ，运行安全性检测算法， $Work = Available = (0, 4, 0, 1)$ ， $Finish[i]=false$ ，此时所有 $Need[i] \leq Work[i]$ 均不成立，结果 $Finish[i]$ 均为 $false$ ，不存在安全进程序列，系统处于不安全状态。系统将取消资源分配并恢复原来状态，进程 $p2$ 等待。

4.证明：假设在一个系统中，有 m 个同类资源，由 n 个进程共享。进程每次只可以申请与释放一个资源。若如下两个条件成立，证明该系统不存在死锁。**a.** 每个进程的最大资源需求量 Max_i 在 1 与 m 之间。**b.** 所有进程的最大需求量之和少于 $m+n$ 。

证明：依据题意 $Max_1 + Max_2 + Max_3 + \dots + Max_n < m+n$ ，若发生死锁，那么所有的资源都应被分配，则有 $allocation_1 + allocation_2 + allocation_3 + \dots + allocation_n < m$ ，因为 $Max_i = need_i + allocation_i$ ；所以由以上两式得 $need_1 + need_2 + \dots + need_n < n$ ；这个式子表明 n 个进程还需要的资源之和小于 n ，这就必然有一个 $need_i = 0$ ，那么这个程序已经获得全部资源被执行，然后释放资源，这样死锁的条件也就不存在了。

七.PV 问题

1. P\|V 操作定义

S 为一个信号量，P、V 操作可描述为：

P(S): while $S \leq 0$ do skip

$S := S - 1$;

V(S): $S := S + 1$;

设有三个并发进程：进程 Reader 负责从输入设备读入信息并传送给进程 Handler，进程 Handler 将信息加工并传送给进程 Printer，进程 Printer 将进行打印输出。其中，三个进程共享同一个缓冲区，且缓冲区大小为 K 。请使用 P/V 操作，写出正确的并发程序。

```

var
empty, full, ok, mutex: semaphore;
inR, outR, inP, outP: integer;
buffer: array 0..k-1 of item;
procedure Reader
begin
while true do
begin
    输入数据 data1;
    P(empty);
    P(mutex);
    buffer(inR) := data1;
    inR := (inR+1) mod (k);
    V(mutex);

```

```

    V(full);
end
end
procedure Handler
begin
while true do
begin
    P(full);
    P(mutex);
    data2 := buffer(outR);
    outR := (outR+1) mod (k);
    V(mutex);
    对 data2 加工;
    P(mutex);
    buffer(inP) := data2;
    inP := (inP+1) mod (k);
    V(mutex);
    V(ok);
end
end
procedure Printer
begin
while true do

```

```

while true do
begin
    P(ok);
    P(mutex);
    data3 := buffer(outP);
    outP := (outP+1) mod (k);
    V(mutex);
    V(empty);
    打印 data3;
end
end
begin
seminitial(empty.v,k; full.v,0; ok.v, 0; mutex.v,1);
inR:=0; outR:=0;
inP:=0; outP:=0;
cobegin
    Printer;
    Handler;
    Printer;
coend
end

```

八. 存储管理

1. 设某计算机的逻辑地址空间和物理地址空间均为 64KB，按字节编址。操作系统最多为一个进程分配 4 页物理内存，页的大小为 1KB，并采用固定分配局部置换策略。在时刻 260 前，某进程内存分配与访问情况如下表所示

页号	页框号	装入时间	访问时间
0	7	130	250
1	4	230	230
2	2	200	240
3	9	160	245

答：(1) 17CAH 转换为二进制为：0001 0111 1100 1010，页的大小为 1KB，所以页内偏移为 10 位，于是前 6 位是页号，所以其页号为 0001 01，转换为 10 进制为 5，所以，17CA 对应的页号为 5。

(2) 若采用先进先出置换算法，则被置换出的页号对应的页框号是 7，因此对应的二进制物理地址为：0001 1111 1100 1010，转换为 16 进制位的物理地址为 1FCAH。

(3) 若采用 LRU，应该置换的页框号是 4，因此对应的二进制物理地址为：0001 0011 1100 1010，转换为 16 进制物理地址为 13CAH。

2. 某计算机提供给用户 2^{32} 字节的虚拟存储空间，虚拟存储器采用一级页表实现，页面大小是 4K 字节。某进程的页表内容如下表所示，操作系统最多为进程分配 2 页物理内存，采用最近最少使用置换算法(LRU)和局部淘汰策略。设又虚地址访问序列 2111H、191AH、2315H，请进程页表占用多少内存空间？请说明理由。(5 分)

191AH 的物理地址是多少？请说明理由。(10 分)

页号	页框号（物理块号）	特征位（存在位）
0	10H	1
1		0
2	41H	1

答：（1）4MB

（2）因为页面大小 4K 字节，页内偏移占 12 位，虚地址 191AH 被分成两部分，页号占 4 位，P=1，页内偏移 D=91AH。由于进程工作集为 2，需要替换第 0 页，因此 191AH 的对应的物理块号为 10H。物理地址为 $10H \times 4K + 91AH = 1091AH$ 。

3. 设一个逻辑地址空间有 8 个页面，每页大小为 1024B，先将它映象到 32 块物理块的内存上。试问：

（1）逻辑地址要用多少位表示？

（2）物理地址要用多少位表示？

答：（1）因为逻辑页号 8 页，故需要 3 位表示，而块大小和页大小相等，故，块内地址需要 10 位。因此，逻辑地址需要 13 位表示。

（2）内存有 32 块，故需要 5 位二进制数表示，而块大小和页大小相等，故块内地址需要 10 位表示。因此，物理地址需要 $5+10=15$ 位。

九.并发问题

下面是两个并发执行的进程。它们能正确运行吗？若不能请举例说明，并改正之：

```
cobegin
  var x: integer;
  procedure P1
    var y, z: integer;
    begin
      x: =1;
      y: =0;
      if x≥1 then y: =y+1;
      z: =y;
    end
  end
  procedure P2
    var t, u: integer;
    begin
      x: =0;
      t: =0;
      if x≤1 then t: =t+2;
      u: =t;
    end
  end
coend
```

答：不能正确运行。例如：先执行完整个 P1，再执行 P2，那么 P1 中 y 的值为 1。但是如果执行到 P1: x:=1;时，切换到 P2 执行，然后再执行 P1，那么那么 P1 中 y 的值为 0。同样条件的两次运行，其结果是不确定的。

有很多种改正方法，下面是一个例子。

```
cobegin
  var empty: semaphore := 0;
  var x: integer;
  procedure P1
    var y, z: integer;
    begin
      P(empty);
      x: =1;
      y: =0;
      if x≥1 then y: =y+1;
      z: =y;
    end
  end
  procedure P2
    var t, u: integer;
    begin
      P(empty);
      x: =0;
      t: =0;
      if x≤1 then t: =t+2;
      u: =t;
    end
  end
coend
```

```
      x: =1;  
      y: =0;  
      if  $x \geq 1$  then y: =y+1;  
      z: =y;  
      V(empty);  
    end  
  coend
```

```
      x: =0;  
      t: =0;  
      if  $x \leq 1$  then t: =t+2;  
      u: =t;  
    end
```