

# 作业十 操作系统接口

## 一、填空题

1、常见的操作系统接口方式有（命令）、（图形化接口）、（系统调用）

## 二、回答题

1、什么是系统调用？系统调用和一般的过程调用有什么异同？

（1）系统调用：系统调用提供了用户程序与操作系统之间的接口，是运行在用户空间的应用程序向操作系统内核请求某些服务的调用过程。应用程序利用系统调用，在程序一级动态请求和释放系统资源，调用系统中已有的系统功能来完成哪些与机器硬件部分相关的工作以及控制程序的执行速度等。

（2）系统调用和库函数的区别：

a.库函数是语言或应用程序的一部分，系统调用是内核提供给应用程序的接口，是系统程序的一部分。

b.库函数在用户地址空间执行，系统调用在内核地址空间执行。

c.库函数调用属于过程调用，开销较小；系统调用需要在用户空间和内核上下文环境间切换，开销比较大。

d.系统调用依赖于平台，库函数不依赖。

2、用户程序在需要 OS 提供某种服务时，是通过系统调用来完成的。请以一个具体例子（如读写磁盘、在显示屏幕上显示字符等）说明系统调用的处理过程。你可以按照一个你熟悉的操作系统（如 UNIX、WINDOWS、LINUX）来说明，也可以介绍你自己根据某个硬件环境设计的系统调用的处理过程。

系统调用的基本过程：

用户程序执行到系统调用指令（读取磁盘指令）

通过访管指令产生中断（由用户态转到内核态的开始）

系统进入访管中断处理

保护现场，并通过系统调用入口表，按功能号跳转找到相应功能入口地址

执行相应例行程序（在内核态中执行磁盘读取指令），结束后正常情况返回系统调用的下一条指令执行（由内核态重新回到用户态）

## 作业 2：进程管理-进程概念及进程控制

### 一、选择与填空

- 下面对进程的描述中，不正确的是（ ）。  
A.进程是动态的概念    B.进程执行需要处理机  
C.进程是有生命周期的    **D.进程是指令的集合**
- 下面的进程状态变化中，（ ）变化是不可能发生的。  
A.运行→就绪    B.运行→等待  
**C.等待→运行**    D.等待→就绪
- 分配到必要的资源并获得处理机时的进程状态是（ ）。  
A.就绪状态    **B.执行状态**  
C.阻塞状态    D.撤销状态
- 进程的结构特征是指进程由三个部分组成，分别是程序、\_\_数据\_\_和\_\_PCB\_\_；
- 进程的特征有\_\_\_\_、\_\_\_\_、\_\_\_\_、制约性、\_\_\_\_和异步性。**(动态性、并发性、独立性、制约性、异步性、结构特征)**

### 二、简答题

- 试比较进程和程序的异同？

答：(1)进程是程序的一次运行过程，是一个动态实体，而程序是一个指令的集合，是静态实体；(2)进程具有生命周期，具有创建、执行和撤销的过程，而程序一旦创建，可以永远存在；(3)进程实体由程序段、数据段及进程控制块组成；(4)进程与程序之间不存在一一对应的关系，不同的进程可以对应相同的程序，一个进程中还可以同时调用多个程序；(5)进程实体是一个能独立运行的基本单位，可独立获得资源和独立调度；而程序不能作为独立的单位参加运行；(6)进程可按异步方式运行，程序不是运行实体，所以不可以异步执行。

- 试说明 PCB 的作用，为什么说 PCB 是进程存在的唯一标志？

答：(1)进程控制块 PCB 包含了有关进程的描述信息、控制信息以及资源信息，是进程动态特征的集中反映(2)系统根据 PCB 感知进程的存在和通过 PCB 中所包含的各项变量的变化，掌握进程所处的状态以达到控制进程活动的目的。(3)进程的 PCB 是系统感知进程的唯一实体，常驻内存。

- 进程的三个基本状态及进程状态转换的典型原因？

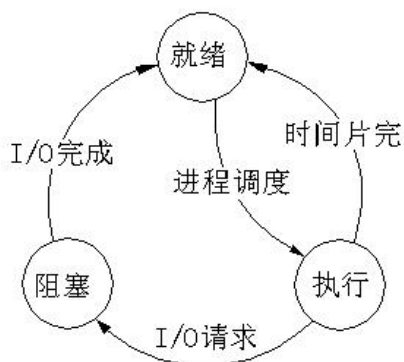


图2-1 进程的三种基本状态及其转换 文字说明略

4、什么是原语？什么时候需要调度进程创建原语？

把系统态下执行的某些具有特定功能的程序段称为原语，原语有指令级和功能级的（程序段）。这两类原语都在系统态下执行，完成某个系统管理所需要的功能和被高层软件调用。原语的执行具有不可分割性。

5、一个状态转换的发生，是否一定导致另一个状态转换发生，列出所有的可能。不一定。

1) 一个处于运行态的进程从运行态转换到就绪或者阻塞或者终止状态，此时就绪队列中如有就绪进程，会导致其中某个就绪进程从就绪状态转为运行态)

2) 一个处于阻塞状态的进程从阻塞状态转换为就绪状态，或者是新创建了一个进程，此时如果 CPU 空闲，会导致该进程从就绪态转换到运行态。或者如果进程调度采用可剥夺方式，如果此进程的优先级高于当前处于运行态的进程，会导致原来运行态的进程由运行转为就绪，此进程由就绪转为运行。

### **思考题：**

日常生活中类似进程的例子。

## 作业四：进程同步/互斥&进程通信与线程

### 一、选择题

- 任何两个并发进程之间存在着（ ）的关系。  
A. 各自完全独立  
B. 拥有共享变量  
C. 必须互斥  
D. 可能相互制约
- 并发进程执行时可能会出现“与时间有关的错误”，这种错误是由于并发进程（ ）引起的。  
A. 使用共享互斥资源  
B. 执行的顺序性  
C. 要求计算时间的长短  
D. 程序的长度
- 用来实现进程同步与互斥的 PV 操作实际上是由（ ）过程组成的。  
A. 一个可被中断的  
B. 一个不可被中断的  
C. 两个可被中断的  
D. 两个不可被中断的
- 进程从运行态变为等待态可能由于（ ）。  
A. 执行了 V 操作  
B. 执行了 P 操作  
C. 时间片用完  
D. 有高优先级进程就绪
- 用 PV 操作管理互斥使用的资源时，信号量的初值应定义为（ ）。  
A. 任意正整数  
B. 1  
C. 0  
D. -1
- 现有  $n$  个具有相关临界区的并发进程，如果某进程调用 P 操作后变为等待状态，则调用 P 操作时信号量的值必定为（ ）。  
A.  $\leq 0$   
B. 1  
C.  $n-1$   
D.  $n$
- 用 PV 操作管理临界区时把信号量的初值定义为 1，现已有一个进程在临界区，但有  $n$  个进程在等待进入临界区，这时信号量的值为（ ）。  
A. -1  
B. 1  
C.  $-n$   
D.  $n$
- 用 V 操作唤醒一个等待进程时，被唤醒进程的状态应变成（ ）状态。  
A. 执行  
B. 就绪  
C. 运行  
D. 收容
- 进程间的同步是指进程间在逻辑上的相互（ ）关系。  
A. 间接  
B. 制约  
C. 继续  
D. 调用
- 对于两个并发进程，设互斥信号量为 mutex，若  $\text{mutex}=0$ ，则\_\_\_\_\_

- A. 表示没有进程进入临界区    B. 表示有一个进程进入临界区
- C. 表示有一个进程进入临界区，另一个进程等待进入
- D. 表示有两个进程进入临界区

11. 设系统中有  $n$  ( $n > 2$ ) 进程，且当前不在执行进程调度程序，试考虑下述 4 种情况哪种不能发生：

- A. 没有运行进程，有 2 个就绪进程， $n-2$  个进程处于等待状态。
- B. 有 1 个运行进程，没有就绪进程， $n-1$  个进程处于等待状态
- C. 有 1 个运行进程，有 1 个就绪进程， $n-2$  个进程处于等待状态
- D. 有 1 个运行进程，有  $n-1$  个就绪进程，没有进程处于等待状态

12. 有  $m$  个进程共享同一临界资源，若使用信号量机制实现对资源的互斥访问，则信号量值的变化范围是\_\_\_\_\_  $[1-m, 1]$  \_\_\_\_\_

13. 设与某资源关联的信号量初值为 3，当前值为 1。若  $M$  表示该资源的可用个数， $N$  表示等待该资源的进程数，则  $M$ 、 $N$  分别是\_\_1, 0\_\_

## 二、设计题

1、三个进程  $P_1$ 、 $P_2$ 、 $P_3$  互斥使用一个包含  $N$  ( $N > 0$ ) 个单元的缓冲区。 $P_1$  每次用 `produce()` 生成一个正整数并用 `put()` 送入缓冲区某个单元中； $P_2$  每次用 `getodd()` 从缓冲区中取出一个奇数并用 `countodd()` 统计奇数个数； $P_3$  每次用 `geteven()` 从缓冲区中取出一个偶数并用 `counteven()` 统计偶数个数。要求某时刻仅能有一个进程对缓冲区进行操作。请用信号量机制实现这三个进程的同步与互斥活动，并说明所定义的信号量的含义。要求用伪代码描述。

```
int buf;
semaphore empty, S12, S13, mutex;
empty.value=N, S12.value=0, S13.value=0, mutex.value=1;
process p1()
{int num;
while (true)
    {num=produce();
    P(empty)
    P(mutex)
    buf=put(num);
    V(mutex)
    if (num MOD 2==0)
        V(S12);
    else if (num MOD 2==1)
        V(S13);
    }
}
process p2()
{int odd;
while (true)
    {P(S12) ;
    P(mutex)
    odd=getodd(buf);
    V(mutex)
    V(empty) ;
    countodd()
    }
}
process p3()
{int even;
while (true)
    {P(S13) ;
    P(mutex)
```

```

        even=geteven(buf);
        V(mutex);
        V(empty) ;
        counteven();
    }

main()
{cobegin
    process p1();
    process p2();
    process p3();
coend
}

```

2 某寺庙有大、小和尚若干，有一水缸。由小和尚挑水入缸供老、大和尚饮用。水缸可容 10 桶水，水取自同一井。水井很窄，每次只能容一个水桶取水。水桶总数为 3 个。每次入、取缸水仅为 1 桶，且不可同时进行。试给出取水、入水的同步算法

**semaphore mutex1, mutex2, empty, full, count;**

**mutex1.value=1; mutex2.value=1;empty.value=10; full.value=0; count.value=3;**

**process 小和尚 Pi():**

```

{
while(true)
{wait(empty);
 wait(count);
 wait(mutex1);
 从井中取水;
 signal(mutex1);
 wait(mutex2);
 送水入水缸;
 signal(mutex2);
 signal(count);
 signal(full);
}
}

```

**process 老和尚 Pj():**

```

{
while(true)
{ wait(full);
 wait(count);
 wait(mutex2);
 从缸中取水;
 signal(mutex2);
 signal(empty);
 signal(count);
}
}

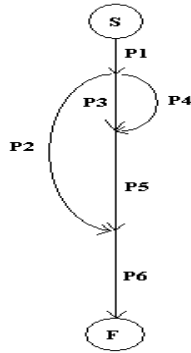
```

```

main()
{cobegin
    P1(),P2(),.....
    P1(),P2(),.....
coend
}

```

3、如图，试用信号量实现这 6 个进程的同步。



semaphore P12=1,p13=1,p14=1,p35=1,p45=1,p26=1,p56=1;

P12()

```
{ process p1...;
  signal(p12); signal(p13); signal(p14);
}
```

P2()

```
{ wait(P12); process p2....;  signal(p26);
}
```

P3()

```
{ wait(P13); process p3....; signal(p35);
}
```

P4()

```
{ wait(P14); process p4....; signal(p45);
}
```

P5()

```
{ wait(P35); wait(P45); process p5....; signal(p56);}
```

P6()

```
{ wait(P26); wait(P56); process p6...; }
```

main()

```
{begin
  P1();
  cobegin
    P2();
    begin
      cobegin
        P3();P4();
      coend
      P5();
    end
  coend
  P6();
end
}
```

4、一条河上架设了由若干个桥墩组成的一座桥。若一个桥墩只能站一个人，过河的人只能沿着桥向前走而不能向后退。过河时，只要对岸无人过，就可以过。但不允许河对岸的两个人同时过，以防止出现死锁。请给出两个方向的人顺利过河的同步算法。

**int countA=0,countB=0;** //将过桥的两个方向分别标记为 A 和 B; 分别表示 A 和 B 方向上已在桥上的行人数;

**semaphore SA, SB, mutex;** //表示对 countA、countB 互斥访问及两方向行人对桥的互斥使用  
**SA.value=1,SB.value=1,mutex.value=1;**

**PA<sub>i</sub>()** //i=1,2,...

```
{ wait(SA);
    if (countA==0) then wait(mutex);
    countA:=countA+1;
    signal(SA);
    通过桥;
    wait(SA);
    countA:=countA-1;
    if (countA=0) then signal(mutex);
    signal(SA);
}
```

**PB<sub>j</sub>()** //j=1,2,....

```
{ wait(SB);
    if (countB=0) then wait(mutex);
    countB:=countB+1;
    signal(SB);
    通过桥;
    wait(SB);
    countB:=countB-1;
    if (countB=0) then signal(mutex);
    signal(SB);
}
```

**main()**

**{Cobegin**

**PA<sub>1</sub>()**,PA<sub>2</sub>().....

**PB<sub>1</sub>()**,PB<sub>2</sub>().....

**CoEnd**

**}**



# 作业五：进程通信与线程

## 一、问答题

### 1、进程通信有哪几种常用方式？

进程通信（inter process communication）就是进程间的数据交换，PV 操作是低级通信方式，高级通信方式是指以较高效率传递大量数据的通信方式。通信模型上分为直接通信和间接通信模型。高级通信方式主要可分为共享通信，消息传递和管道通信三大类，这三大类又可以分为无名管道，有名管道，信号，消息队列，共享内存，信号量，以及套接字七小类。

（1）无名管道（Pipe）：管道可用于具有亲缘关系进程间的通信，允许一个进程和另一个与它有共同祖先的进程之间进行通信。

（2）有名管道（named pipe）：命名管道克服了管道没有名字的限制，因此，除具有管道所具有的功能外，它还允许无亲缘关系进程间的通信。命名管道在文件系统中具有对应的文件名。

（3）信号（Signal）：信号是比较复杂的通信方式，用于通知接受进程有某种事件发生，除了用于进程间通信外，进程还可以发送信号给进程本身；

（4）消息（Message）队列：消息队列是消息的链接表。有足够权限的进程可以向队列中添加消息，被赋予读权限的进程则可以读走队列中的消息。消息队列克服了信号承载信息量少，管道只能承载无格式字节流以及缓冲区大小受限等缺点。

（5）共享内存：使得多个进程可以访问同一块内存空间，是最快的可用 IPC 形式。是针对其他通信机制运行效率较低而设计的。往往与其它通信机制，如信号量结合使用，来达到进程间的同步及互斥。

（6）信号量（semaphore）：主要作为进程间以及同一进程不同线程之间的同步手段。

（7）套接字（Socket）：更为一般的进程间通信机制，可用于不同机器之间的进程间通信。

### 2、试从系统资源管理、处理机调度、资源利用率、优势等几个方面对比说明下进程与线程的区别？

解答要点：在有进程和线程的系统中：

(1)进程是资源管理的基本单位，它拥有自己的地址空间和各种资源，例如：内存空间、外部设备等；

(2)线程是处理机调度的基本单位，它和其他线程一起共享进程的资源。

(3)进程的调度、同步等控制大多由操作系统内核完成；线程的控制既可以由操作系统进行（比如核心级线程），也可以由用户程序控制执行（用户级线程）。

(4)以进程为单位进行处理机切换和调度时，由于涉及到资源转移以及现场保护等问题，将导致处理机切换时间变长，资源利用率降低。以线程为单位进行处理机调度和切换时，由于不发生资源的变化，特别是地址空间的变化，处理机切换时间较短，处理机效率提高。

(5)线程和进程一样，都有自己的状态和同步机制。由于线程没有单独的数据和程序空间，线程不能像进程的数据和程序那样，交换到外存空间，线程没有挂起状态。

(6)多线程系统中，使用线程可以提高进程内任务执行的并发度，减少用户等待时间，提高系统响应时间。

## 作业五： 处理机调度与死锁

### 一、单项选择题

1. 为了根据进程的紧迫性做进程调度，应采用( B )。  
A. 先来先服务调度算法      B. **优先数调度算法**  
C. 时间片轮转调度法      D. 分级调度算法
2. 采用时间片轮转法调度是为了( A )。  
A. **多个终端都能得到系统的及时响应**      B. 先来先服务  
C. 优先数高的进程先使用处理器      D. 紧急事件优先处理
3. 采用优先数调度算法时，对那些具有相同优先数的进程再按( A )的次序分配处理器。  
A. **先来先服务**      B. 时间片轮转  
C. 运行时间长短      D. 使用外围设备多少
4. 当一进程运行时，系统强行将其撤下，让另一个更高优先数的进程占用处理器，这种调度方式是( B )。  
A. 非抢占方式      B. **抢占方式**  
C. 中断方式      D. 查询方式
5. ( B )必定会引起进程切换。  
A. 一个进程被创建后进入就绪态      B. **一个进程从运行态变成阻塞态**  
C. 一个进程从阻塞态变成就绪态      D. 以上都可以
6. ( B )只考虑用户估计的计算机时间，可能使计算时间长的作业等待太久。  
A. 先来先服务算法      B. **计算时间短的作业优先算法**  
C. 响应比最高者优先算法      D. 优先数算法
7. 先来先服务算法以( )去选作业，可能会使计算时间短的作业等待时间过长。
8. 下面所述步骤中，( A )不是创建进程所需的步骤？  
A. **由 CPU 调度程序为进程调度 CPU**      B. 建立一个进程控制块  
C. 为进程分配内存      D. 将进程控制块链入就绪队列
9. 作业在系统中存在与否的唯一标志是( )  
A. 源程序      B. **作业控制块**      C. 作业说明      D. 目的程序
10. 作业进入完成状态时，操作系统( A )  
A. **收回该作业所占的全部资源，消除有关的 PCB 和 JCB，并输出结果**  
B. 将删除该作业，收回该作业所占的全部资源，并输出结果  
C. 将输出结果，并删除该内存中的作业  
D. 收回该作业所占的全部资源，并输出结果
11. 下列哪一项不可能是 CPU 在不可剥夺方式下引起进程调度的原因?( D )  
A. 正在执行的过程调用阻塞原语将自己阻塞起来进入等待状态  
B. 正在执行进程中提出 I / O 请求被阻塞  
C. 正在执行的进程用 P 原语操作，从而因资源不足引起阻塞；或调用 V 源与操作激活等待资源的进程队列  
D. **就绪队列的某个进程的优先级高于当前运行进程的优先级**
12. ( C )是指把作业提交到系统完成的时间间隔：  
A. 响应时间      B. 等待时间      C. 周转时间      D. 运行时间
13. 死锁预防是保证系统不进入死锁状态的静态策略，其解决办法是破坏产生死

锁的四个必要条件之一。下列方法中哪是一个破坏了“循环等待”条件( )

- A. 银行家算法
- B. 资源有序分配策略
- C. 剥夺资源法
- D. 一次性分配策略

14. 一种既有利于短作业又兼顾长期作业的作业调度方式是( D )

- A. 先来先服务
- B. 均衡调度
- C. 最短作业优先
- D. 最高响应比优先

15. 通常不采用( D )方法来解除死锁。

- A. 终止一个死锁进程
- B. 终止所有死锁进程
- C. 从死锁进程处抢夺资源
- D. 从非死锁进程处抢夺资源

## 二、回答题

1、计算机系统中存在着哪几种可能的调度？各自的主要工作是什么？

2、进程调度的时机有哪些？

3、作业调度的性能评价标准有哪些？请结合批处理系统、分时系统、实时系统的特点，说明这几个系统在作业调度性能评价上有什么不同的侧重点和需求？

1、主要有高级调度、中级调度、低级调度

(1) 高级调度又称作业调度，其主要任务是按一定的原则对外存输入井上的大量后备作业进行选择，给选出的作业分配内存、输入输出设备等必要的资源，并建立相应的进程，以使该作业的进程获得竞争处理机的权利。另外，当该作业执行完毕时，还负责回收系统资源。

(2) 低级调度又称为进程调度，其主要任务是按照某种策略和方法选取一个处于就绪状态的进程占用处理机。在确定了占用处理机的进程后，系统必须进行进程上下文切换以建立与占用处理机进程相适应的执行环境。

(3) 中级调度又称交换调度。其主要任务是按照给定的原则和策略，将处于外存交换区中的就绪状态或就绪等待状态的进程调入内存，或把处于内存就绪状态或内存等待状态的进程交换到外存交换区。

2、进程调度的时机

(1)正在执行的进程执行完毕

(2)执行中的进程因为某种原因从运行状态转入阻塞状态（如：自己调用阻塞原语将自己阻塞起来进入睡眠等待状态。执行中进程调用了P原语操作，从而因资源不足而被阻塞，执行中进程提出I/O请求后被阻塞）

(3)分时系统中时间片用完

(4)在CPU执行方式是可剥夺时，就绪队列中某进程的优先级高于当前执行进程的优先级时将引起进程调度。

(5)在执行完系统调用等系统程序后返回用户程序时，可看做系统进程执行完毕，从而调度选择一个新的用户程序执行

3、

(1)作业调度性能评价标准：平均周转时间/平均带权周转时间，平均响应时间，吞吐量、平均等待时间、CPU利用率

(2)对于批处理系统，由于主要用于计算，因而对于作业的周转时间要求较高。作业的平均周转时间或平均带权周转时间被用来衡量调度程序的优劣。

(3)对于分时系统，除了要保证系统吞吐量，资源利用率高之外，还应保证用户能够容忍的响应时间。所以分时系统中，仅用周转时间来衡量调度性能是不够的，平均响应时间是衡量调度性能的一个重要指标。

(4)实时系统中，任务的执行往往有截止时间要求，因此实时系统中满足用户的时限时间是一个重要指标。

### 三、计算题

1. 能同时容纳两道作业的系统，有 5 个作业 P1, P2, P3, P4, P5 均在 0 时刻到达，到达的先后顺序是 P1, P2, P3, P4, P5，预计它们的运行时间为 12, 6, 2, 4, 8min，其优先级分别为 3, 5, 2, 1, 4，这里数字越小优先级越高，1 为最高优先级。

要求：分别写出作业调度采用短作业优先调度算法和优先级调度算法，进程调度采用时间片轮转调度算法（时间片为 2min），作业调度顺序各是多少？

略

2、根据下表给出的进程调度信息，试用 Gantt 图(甘特图)描述分别采用 FCFS、SJF 调度算法、非抢占式优先权、抢占式优先权及时间片轮转（时间片为 1）算法的执行次序，并计算平均周转时间和平均等待时间。（注：优先数越大，优先权越小）

进程	到达时刻（秒）	执行时间	优先数
P1	0.0	10	3
P2	2.0	1	1
P3	4.0	2	3
P4	5.0	1	4
P5	6.0	5	2

回答：（Gantt 图略）

FCFS 算法 平均等待时间=31/5 平均周转时间=50/5=10

处理的作业	到达时间	开始时间	结束时间	等待时间	周转时间
P1	0	0	10	0	10
P2	2	10	11	8	9
P3	4	11	13	7	9
P4	5	13	14	8	9
P5	6	14	19	8	13

SJF 算法 平均等待时间=30/5=6 平均周转时间=49/5

处理的作业	到达时间	开始时间	结束时间	等待时间	周转时间
P1	0	0	10	0	10
P2	2	10	11	8	9
P4	5	11	12	6	7
P3	4	12	14	8	10
P5	6	14	19	8	13

非抢占式优先权算法 平均等待时间=38/5 平均周转时间=57/5

处理的作业	到达时间	开始时间	结束时间	等待时间	周转时间
P1	0	0	10	0	10
P2	2	10	11	8	9
P5	6	11	16	5	10

P3	4	16	18	12	14
P4	5	18	19	13	14

3. 用银行家算法考虑下列系统状态：

进程	分配矩阵	最大需求矩阵	资源总数矩阵
A	3 0 1 1	4 1 1 1	6 3 4 2
B	0 1 0 0	0 2 1 2	
C	1 1 1 0	4 2 1 0	
D	1 1 0 1	1 1 1 1	
E	0 0 0 0	2 1 1 0	

问：

- (1) 此时系统是否安全？为什么？
- (2) 若进程 B 请求 (0, 0, 1, 0)，可否立即分配？
- (3) 此后进程 E 也请求 (0, 0, 1, 0)，可否分配给它？

回答：

- 1、安全：存在 DEABC 资源安全分配序列，
- 2、可以，分配后，仍然可以按照 DEABC 的序列完成资源的安全分配
- 3、不可以。

## 作业七： 存储器管理 1

### 一、选择题

- 1、动态分区管理要求对每一个作业都分配（ ）的内存单元。  
A. 地址连续  
B. 若干地址不连续  
C. 若干连续的块  
D. 若干不连续的块
- 2、可变分区管理中的紧凑技术可以（ ）。  
A. 集中空闲分区  
B. 增加内存容量  
C. 缩短访问周期  
D. 加速地址转换
- 3、可变分区管理中采用的地址转换公式（ ）。  
A. 绝对地址=界限寄存器值+逻辑地址  
B. 绝对地址=上界寄存器值+逻辑地址  
C. 绝对地址=基址寄存器值+逻辑地址  
D. 绝对地址=块号\*块长+逻辑地址
- 4、可变分区管理中，（ ）分配作业的主存空间。  
A. 根据一个空闲块链  
B. 根据空闲分区表  
C. 根据位示图表构成的准粗分配表  
D. 系统自由分配
- 5、可变分区管理中，某一作业完成后，系统收回其内存空间，并与相邻空闲去合并，为此修改空闲分区表，造成空闲分区数减1的情况是（ ）  
A. 无上邻空闲分区，也无下邻空闲分区  
B. 有上邻空闲分区，但无下邻空闲区  
C. 有下邻空闲分区，但无上邻空闲区  
D. 有上邻空闲分区，也有下邻空闲分区
- 6、首次适应算法的空闲分区是（ ），最佳适应算法的空闲分区是（ ）  
A. 按大小递减顺序连在一起  
B. 按大小递增顺序连在一起  
C. 按地址由小到大排列  
D. 按地址有大到小排列
- 7、下列内存分配算法中，（ ）更易产生无法利用的小碎片  
A. 单一连续分配  
B. 固定分区分配  
C. 可变分区分配  
D. 页式分配
- 8、（待定）段式和页式存储管理的地址结构很类似，但是它们有实质上的不同，以下正确的是（ ）  
A. 页式的逻辑地址是连续的，段式的逻辑地址可以不连续  
B. 页式的地址是一维的，段式的地址是二维的

- C. 分页是操作系统进行的，分段是用户确定的  
D. 页式采用静态重定位方式，段式采用动态重定位方式
- 9、段页式管理中，地址映像表是（ ）。  
A. 每个进程一张段表，一张页表      B. 进程的每个段一张段表，一张页表  
C. 每个进程一张段表，每个段一张页表      D. 每个进程一张页表，每个段一张段表
10. 离散存储管理的主要特点是（ ）。  
A. 不要求将作业装入到内存的连续区域  
B. 不要求将作业同时全部装入到内存的连续区域  
C. 不要求进行缺页中断处理  
D. 不要求进行页面置换
- 11、某页式存储管理系统中，地址寄存器长度为 24 位，其中页号占 14 位，则主存的分块大小是（ ）字节  
A.  $2^{10}$       B. 10      C.  $2^{14}$       D.  $2^{24}$
- 12 、在无快表的段页式存储管理系统中，为获得一条指令需要访问内存( )次  
A. 1      B. 2      C. 3      D. 4

## 二、简答

1、什么是逻辑地址？物理地址？地址重定位？实现地址重定位的方法有几类？

【解答】

(1)逻辑地址转换为物理地址的过程称为地址重定位。主要有静态地址重定位和动态地址重定位。

(2)静态地址重定位是在虚空间程序执行之前由装入程序完成地址映射过程。动态地址重定位是在程序执行过程中,在 CPU 访问内存之前由硬件地址变换机构将要访问的程序或数据的逻辑地址转换为内存物理地址。

2、什么是内部碎片和外部碎片？常见的内存分配技术中都会产生什么碎片？

【解答】

(1)内部碎片是指容量很小并且不能再被分配的内存区域；外部碎片是指容量很小但是还能再被分配的内存区域

(2)单一连续分配：内部碎片；

固定分区分配：内部碎片；

可变分区分配：外部碎片；

页式管理：内部碎片；

段式：外部碎片；

段页式：内部碎片；

3、连续存储和离散存储的比较。

【解答】

1、连续存储是指一个用户程序分配一个连续的内存空间，又称分区管理方式，主要包括固定分区、可变分区。离散存储是给用户程序分配不连续的内存空间，主要包括页式、段式、段页式。

2、连续存储管理相对简单，重定位计算简单，但内存利用率不高，不适合程序空间的动态增长。离散存储管理相对复杂，重定位计算复杂，需要硬件支持，内存利用率高，适合用户程序的动态增长。

## 三、计算题



1、已知主存容量为 512KB，其中操作系统代码占低址部分 126KB，有作业序列如下：

作业 1 要求 80KB；

作业 2 要求 56KB；

作业 3 要求 120KB；

作业 1 完成

作业 3 完成

作业 4 要求 156KB；

作业 5 要求 80KB；

试用最佳适应算法处理上述作业序列；并做以下工作：

(1) 画出作业 1, 2, 3 进入系统后的内存分布和空闲分区表的情况；

(2) 画出作业 1, 3 完成后内存分布和空闲分区表的情况；

(3) 画出作业 4, 5 进入系统后的内存分布和空闲分区表的情况；

126K	126K	126K
80K(J1)	80K(空闲)	80K(J5)
56K(J2)	56K(J2)	56K(J2)
120K(J3)	250K (空闲)	156K(J4)
130K (空闲)		94K (空闲)

2、在采用分页存储管理系统中，地址结构长度为 18 位，其中 11 至 17 位表示页号，0 到 10 位表示页内位移。若有一作业的各页依次放入 2, 3, 7 号物理块中，请问：

(1) 主存容量最大可为多少 K？分为多少块？每块多大？

(2) 逻辑地址 1500 应在几号页内？对应的物理地址为多少？

【解答】

(1) 地址总长度为 18 位，所以主存容量为  $2^{18}=256k$  (bytes)

页号用 7 位表示，所以主存分为  $2^7=128$  块

页内地址用 10 位表示，所以每块大小为  $2^{10}=1k$  (bytes)

(2)  $INT(1500/2048)=0$      $MOD(1500/2048)=1500$

所以逻辑地址 1500 在 0 号页内，偏移地址是 1500

所以对应的物理地址为：  $2*2048+1500=5596$

3、设有一页式存储管理系统，向用户提供的逻辑地址空间最大为 16 页，每页 2048B，内存总共有 8 个存储块。试问逻辑地址至少应为多少位？内存空间有多大？

【解答】

逻辑地址空间最大为 16 页。每页 2048B，所以逻辑地址空间大小是  $16*2048=2^{15}$ (bytes)，逻辑地址至少为 15 位。

内存空间是  $8*2048=2^{14}$ (bytes)，内存地址为 14 位。



4、某计算机有 64 位虚地址空间，页大小是 2048B. 每个页表项长为 4B。因为所有页表都必须包含在一页中，故使用多级页表，问一共需要多少级？

**【解答】**

页大小是 2048B，每个页表项长为 4B, 所以每个页内能存放  $2048/4=2^9$  个页表项。

虚地址空间是 64 位，所以虚地址空间大小是  $2^{64}$ ，页大小是  $2048=2^{11}$ B，所以共需  $2^{53}$  个页存储。

因为所有页表都必须包含在一个页面中，所以顶层页表只能在一个页面中， $(2^9)^5 < 2^{53} < (2^9)^6$ ，所以共需 6 级页表。