

1. 下列几种关于进程的叙述，\_\_\_\_最不符合操作系统对进程的理解。

A. 进程是在多程序环境中的完整程序

B. 进程可以由正文段、数据段和进程控制块描述

C. 线程 (Thread) 是一种特殊的进程

D. 进程是程序在一个数据集合上的运行过程，它是系统进行资源分配和调度的一个独立单位

【答案】A

【解析】在这里程序和进程的概念是有区别的，如：程序没 PCB，而进程有。进程除了程序代码以外，还有数据段和进程控制块结构等。

2. 当被阻塞进程所等待的事件出现时，如所需数据到达或者等待的 I/O 操作已完成，则调用唤醒原语操作，将等待该事件的进程唤醒。请问唤醒被阻塞进程的是\_\_\_\_。

A. 父进程            B. 子进程

C. 进程本身        D. 另外的或与被阻塞进程相关的进程

【答案】D

【解析】进程本身处在阻塞状态，当然无法唤醒自己；被阻塞的进程有可能被父进程或子进程唤醒，当然不一定全是，在信号量机制中，被执行 V 操作的进程唤醒。因此选择答案 D 最合适。

3. 并发进程执行时可能会出现“与时间有关的错误”，引起这种错误的原因是\_\_\_\_。

A. 进程执行的顺序性        B. 访问了共享变量

C. 程序的结构                D. 需要的处理器时间

【答案】B

【解析】两个并发进程，其中一个进程对另一个进程的影响常常是不可预期的，甚至无法再现。一个进程的执行可能影响其它进程的执行结果。即使在正确运行的前提下，程序结果也将可能是不确定的，计算过程具有不可再现性。因此，各种与时间有关的错误就可能出现，与时间有关的错误有两种表现形式，一种是结果不唯一；另一种是永远等待。如果没有对访问共享变量实现互斥，并发进程交替执行可能会产生结果的不一致性。

4. 某进程由于需要从磁盘上读入数据而处于等待状态。当系统完成了所需的读盘操作后，此时该进程的状态将\_\_\_\_。

A. 从就绪变为运行

B. 从运行变为就绪

C. 从运行变为阻塞

D. 从等待变为就绪

【答案】D

【解析】当磁盘读数据时，进程需要这些数据进行计算，不能继续运行，此时进程处于等待状态。当磁盘中的数据读入内存后，此进程具备了继续运行的条件了，唤醒此等待的进程，把其状态由等待状态变为就绪状态

5. 在一个只有单处理机（不考虑多核）的操作系统中，进程有运行、就绪、等待三个基本状态。假如某时刻该系统中有 10 个进程并发执行，且 CPU 为非核心态情况下，试问：

1) 这时刻系统中处于运行状态的进程数最多有几个？最少有几个？

2) 这时刻系统中处于就绪状态的进程数最多有几个？最少有几个？

3) 这时刻系统中处于等待状态的进程数最多有几个？最少有几个？

【分析】

1) 因为系统中只有一个处理机，所以某时刻处于运行状态的进程数最多只有一个。而

最少可能为 0，此时其它 10 个进程一定全部排在各等待队列中，在就绪队列中没有进程，在实际的操作系统中，此时 CPU 是在运行操作系统的空闲进程 (System Idle Process) 或线程。

- 2) 处于就绪状态的进程数最多只有 9 个，不可能出现 10 个情况，因为一旦 CPU 有空，调度程序马上调度；处于就绪状态的进程数最少是 0 个，1 个进程运行 9 个进程等待，或者 10 个进程全部等待。
- 3) 处于等待状态的进程数最多有 10 个；等待状态的进程数最少是 0 个。

6. 假设磁头当前位于第 105 道，正在向磁道序号增加的方向移动。现有一个磁道访问请求序列为 35、45、12、68、110、180、170、195，且用 SCAN 调度（电梯调度）算法得到的磁道访问序列是\_\_。
- A. 110、170、180、195、68、45、35、12
  - B. 110、68、45、35、12、170、180、195
  - C. 110、170、180、195、12、35、45、68
  - D. 12、35、45、68、110、170、180、195

【答案】A

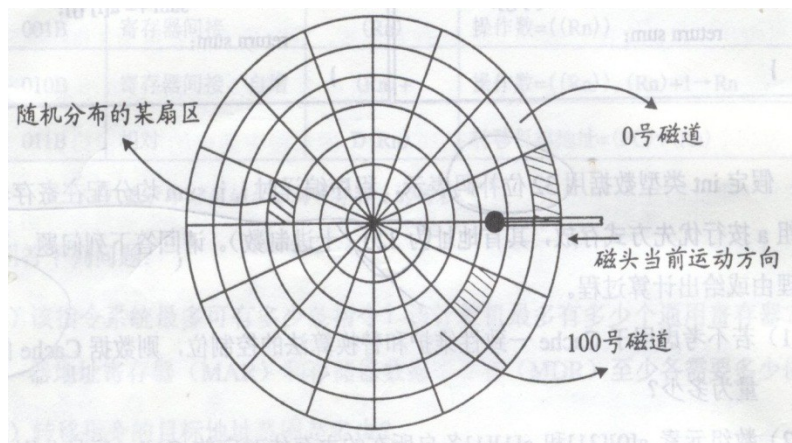
【解析】SCAN 算法的基本思想：磁头从磁盘的一端开始向另一端移动，沿途响应访问请求，直到到达了磁盘的另一端，此时磁头反向移动并继续响应服务请求。根据 SCAN 算法，可以得到访问序列是 A。

7. 对磁盘进行移臂调度时，既考虑了减少寻找时间，又不频繁改变动臂的移动方向的调度算法是\_\_。
- A. 先来先服务
  - B. 最短寻找时间优先
  - C. 电梯调度
  - D. 优先级高者优先

【答案】C

【解析】扫描算法可分为电梯调度 (SCAN) 算法和单向扫描 (CSCAN) 算法。电梯调度 (SCAN) 算法是在磁头前进方向上最短寻找时间的服务，如果前进方向上没有请求（即处理完最高/低编号柱面请求后），则掉转方向。CSCAN 算法很大程度上消除了 SSTF 算法的不公平性，但仍有利于中间磁道的请求。单向扫描 (CSCAN) 算法是对 SCAN 算法的改进，它总是按同一方向移动磁头，当处理完最高编号柱面请求后，不是掉转方向，而是把磁头移动最低编号的柱面请求处，然后按同一方向继续向上移动。这种算法彻底消除了对两端磁道请求的不公平。

8. 假设计算机系统采用 CSCAN（循环扫描）磁盘调度策略，使用 2KB 的内存空间记录 16384 个磁盘块的空闲状态。
- (1) 请说明在上述条件下如何进行磁盘块空闲状态管理。
  - (2) 设某单面磁盘旋转速度为每分钟 6000 转。每个磁道有 100 个扇区，相邻磁道间的平均移动时间为 1ms。若在某时刻，磁头位于 100 号磁道处，并沿着磁道号增大的方向移动（如下图所示），磁道号请求队列为 50、90、30、120，对请求队列中的每个磁道需读取 1 个随机分布的扇区，则读完这 4 个扇区总共需要多少时间？要求给出计算过程。



(3) 如果将磁盘替换为随机访问的 Flash 半导体存储器（如 U 盘、SSD 等），是否有比 CSACN 更高效的磁盘调度策略？若有，给出磁盘调度策略的名称并说明理由；若无，说明理由。

#### 【分析】

本题的知识点：空闲外存储空间的管理方法（考题中的位图表 bitmap），磁盘调度算法（考题中的 CSCAN 算法），磁盘的结构（考题中的平均旋转延时的计算）。这些知识点都包含在操作系统原理的本科教学大纲中，要求考生必须掌握的。笔者在阅卷中了解到，考生这本题的得分少，且得 0 分考生多。在学习操作系统原理时，考生需要充分理解和掌握操作系统的概念、原理和算法，并且能够灵活运用。

#### 【答案】

(1) 用位图表示磁盘的空闲状态。每一位表示一个磁盘块的空闲状态，共需要  $16384/8=2048$  字节=2KB。系统提供的 2KB 内存能正好能表示 16384 个磁盘块。

(2) 采用 CSCAN 调度算法，访问磁道的顺序为 50、90、30、120，则磁头移动磁道长度为  $20+90+20+40=170$ ，总的移动磁道时间为  $170 \times 1\text{ms}=170\text{ms}$ 。

由于转速为 6000 转/分，则平均旋转延迟为  $(60/6000)/2 \text{ s}=5\text{ms}$ ，要访问 4 个磁道，总的旋转延迟时间为  $4 \times 5\text{ms}=20\text{ms}$ 。

由于转速为 6000 转/分，则读取一个磁道上的一个扇区的平均读取时间为  $(60/6000)/100 \text{ s}=0.1\text{ms}$ ，总的读取扇区的时间  $=4 \times 0.1\text{ms}=0.4\text{ms}$ 。

读取上述磁道上所有扇区所花的总时间  $=170\text{ms}+20\text{ms}+0.4\text{ms}=190.4 \text{ ms}$

(3) 采用 FCFS（先来先服务）调度策略更高效。因为 Flash 半导体存储器的物理结构不需要考虑寻道时间和旋转延迟，可直接按 I/O 请求的先后顺序服务。