

3. 答: 存储器映射 I/O 是操作系统中访问 I/O 设备的某种方法, 它将 I/O 设备的寄存器或控制器映射到内存地址空间, 使 I/O 设备可以像该设备一样进行读写。

存储器映射 I/O 可提供统一的编程模型, 提高性能, 简化驱动程序和方便的控制, 适用于需要频繁访问和高性能数据传输的设备。

4. 答: DMA 代表直接内存访问, 允许设备不经 CPU 参与直接访问系统内存, 并直接读取/写入数据到内存。

使用 DMA 可降低 CPU 负载, 高速传输数据, 异步操作, 减少数据拷贝, 适用于需要大量传输数据的场景。

11. 答: (a) 在设备驱动程序层 (b) 在设备无关 I/O 软件层 (c) 在设备驱动程序层, (d) 在设备无关 I/O 软件层, (e) 在应用程序层。

12. 答: 这样做可以减轻主机负载, 提高系统效率, 方便多个打印任务的排队和优先级。在主机故障断电时仍能存储文件内容, 提供容错机制, 支持多用户环境下的任务排队和优先级。

14. 答: 无论是需要 S 还是既需要 S 不需要 R, 都不会引发死锁。因为 C 可以在 A 释放 R₁ 后再运行。

15. 答: 令 D 再申请一个单位, 会导致不安全。此时只剩一个单位可用资源, 而没有一个进程可在使用该资源后达到最大额度。而 C 再申请一单位资源后, 仍可再占用最后一单位资源并达到最大额度, 故不会进入不安全状态。

17. 答: 不会导致死锁。此时: D 先占用一台打印机, 运行完后释放资源 A = (1, 1, 2, 1) 此后, 进程 A: 提供磁带机给扫描仪一台, 运行完后 A = (4, 1, 3, 2) 此后, B, C, E 可以任一顺序运行。

22. 答: 在该算法中, 每次找到一个可执行进程需要在 n 个进程中逐一比较, m 个进程是否满足条件, 故需 mn 次比较。共有 n 个进程要执行, 故一共需要 $mn * n = mn^2$ 次比较。故 a = 1, b = 2。

28. $\frac{R}{b}$. (a). $(10 + 12 + 2 + 18 + 38 + 34 + 32) \times 6 \text{ ms} = 876 \text{ ms}$
(b). $(0 + 2 + 12 + 4 + 4 + 36 + 2) \times 6 \text{ ms} = 360 \text{ ms}$
(c). $(0 + 2 + 16 + 2 + 30 + 4 + 4) \times 6 \text{ ms} = 348 \text{ ms}$