2021013068李天择

平时作业1

答：以多道批处理为开端的现代操作系统通过引入进程的概念，使得多个任务可以并发执行，充分利用了CPU资源，提高了系统的效率和吞吐量。

同步：在多道批处理系统中，多个进程同时执行，彼此之间可能存在依赖关系。同步机制的引入可以确保进程按照预定的顺序执行，避免产生不确定的结果。常见的同步机制包括信号量、互斥量和条件变量等。通过正确使用这些同步机制，可以有效地控制进程的执行顺序，保证系统的正确性和一致性。

互斥：在多道批处理系统中，资源的共享是必不可避免的。在多个进程同时访问临界资源时，可能会引发竞态条件导致数据不一致的问题。为了避免这种情况，需要引入互斥机制，比如临界区、互斥量和条件变量等。通过互斥机制，只有一个进程能够进入临界区或者使用共享资源，其他进程需要等待，确保资源的正确访问。

死锁：死锁是多道批处理系统中常见的问题之一。当多个进程出现循环等待资源的情况时，可能会导致系统陷入死锁状态，进程无法继续执行。为了避免死锁的发生，可以采用一些常见的死锁避免和死锁检测算法，比如银行家算法、资源分配图等。这些算法可以通过合理地资源分配和动态地资源申请释放来防止死锁的发生。

平时作业2：

答：典型的内存分配管理方法主要包括以下几种：

静态分配：在编译或加载阶段，为每个进程分配固定大小的内存空间。静态分配简单高效，但是浪费了内存资源，不够灵活。

动态分配：根据进程的实际需要，在运行时动态地分配内存空间。常见的动态内存分配方法包括堆、栈和伙伴系统等。

分段分页：将内存划分为多个固定大小的段或页，便于管理和分配。分段适用于不规则大小的内存需求，分页适用于规则大小的内存需求。

虚拟内存：通过将虚拟地址空间映射到物理内存，使得进程能够访问比实际物理内存更大的内存空间。虚拟内存可以将磁盘作为扩展的内存使用，提高内存的利用率。

为了提高内存的使用效率，可以采取以下措施：

内存复用：及时释放不再使用的内存空间，以便其他进程或线程使用。

内存压缩和解压缩：对于不常用或者较大的内存块，可以将其压缩存储，待需要时进行解压缩，减少内存占用。

内存回收和碎片整理：及时回收无效的内存块，对碎片化的内存进行整理和合并，以便获得更大且连续的内存空间。

对比硬盘存储空间分配，有以下共性和特性：

共性：

都需要进行合理的空间管理，以提高存储效率和性能。

都可以采用动态分配和静态分配等方法，根据实际需求进行空间分配。

都需要考虑资源的回收和释放，以避免资源的浪费和不足。

特性：

硬盘存储空间相对较大，可以容纳更多的数据，但访问速度相对较慢。

内存分配通常更加频繁和细粒度，而硬盘存储空间分配通常更为稳定和持久。

平时作业3：

答：

操作系统主要管理计算机的以下事务：

1. 进程管理：涉及分配和调度的事务。进程管理负责创建、终止和调度进程，以及为进程提供所需的资源和环境。其中，进程的分配和调度是关键环节。常见的进程调度算法包括：

先来先服务调度：按照进程到达的顺序进行调度，不考虑执行时间。

最短作业优先调度：选择最短的执行时间进行调度，缺点是无法应对长作业的等待时间。

轮转调度：按照时间片轮流分配CPU时间给多个进程。

最高优先级调度：根据进程的优先级进行调度，优先级高的先执行。

2. 存储管理：涉及分配和回收的事务。存储管理负责管理计算机的内存和外存。常见的内存分配算法有：

静态分区分配：将内存划分为固定大小的分区，每个进程分配固定的分区。

动态分区分配：根据进程的实际大小，动态分配内存空间。

页面式分配：将内存划分为固定大小的页，将进程的虚拟地址空间映射到物理内存。

段式分配：将进程划分为多个段，每个段可以具有不同的大小，提高内存的利用效率。

3. 文件系统管理：涉及文件的创建、组织、存储和访问的事务。文件系统管理负责管理文件的物理存储、目录结构、文件权限和访问控制等。常见的文件系统算法有：

文件分配表：使用表格记录文件块的分配情况。

索引节点：使用索引节点来记录文件的元数据和物理块地址。

4. 设备管理：涉及设备分配和调度的事务。设备管理负责管理计算机的硬件设备，包括输入/输出设备和外部存储设备。常见的设备调度算法有：

先来先服务调度：按照请求的顺序进行设备调度。

最短作业优先调度：选择处理时间最短的请求进行设备调度。

涉及虚拟的事务是指虚拟化技术的应用，包括虚拟内存和虚拟机。虚拟内存提供了一种抽象层，将进程的虚拟地址空间映射到物理内存，使得进程能够访问比实际物理内存更大的内存空间。虚拟机则通过在计算机上创建多个虚拟的计算环境，实现共享硬件资源的高效利用，并提供隔离和安全性。

涉及中间件的事务是指操作系统与上层应用之间的中间层，提供方便的接口和服务。例如，数据库管理系统是一种中间件，它提供了数据库的创建、管理和查询等功能。中间件还包括消息队列中间件、缓存中间件和分布式事务中间件等，用于提供高可用性、可靠性和可扩展性的服务。这些中间件可以提供更高层次的抽象和服务，简化应用程序的开发和管理。

内存的空间利用率相对高，而硬盘存储通常存在一定的碎片化和浪费。