面试真题

【腾讯】进程和线程的区别

**进程**：

是资源分配的基本单位，一个进程可以包含多个线程

进程有独立的上下文（代码、数据、文件）等，创建和切换开销大

进程之间是相互独立的，需要进程间通信（IPC）机制，如管道、消息队列、共享内存、套接字等

线程：

是CPU 调度的基本单位，属于进程

线程组内共享资源，但有独立的PC和堆栈，创建和切换开销较小

线程直接读写内存即可，但需要同步机制以避免数据竞争

为什么进程崩溃不会对其他进程产生很大影响

进程间不共享文件和网络连接等，崩溃时内存空间被操作系统回收。

面向简历

进程、线程、虚拟内存

**进程**：

是资源分配的基本单位，一个进程可以包含多个线程

进程有独立的上下文（代码、数据、文件）等，创建和切换开销大

进程之间是相互独立的，需要进程间通信（IPC）机制，如管道、消息队列、共享内存、套接字等

线程：

是CPU调度的基本单位，属于进程

线程组内共享资源，但有独立的PC和堆栈，创建和切换开销较小

线程直接读写内存即可，但需要同步机制以避免数据竞争

虚拟内存

不常用数据暂存硬盘

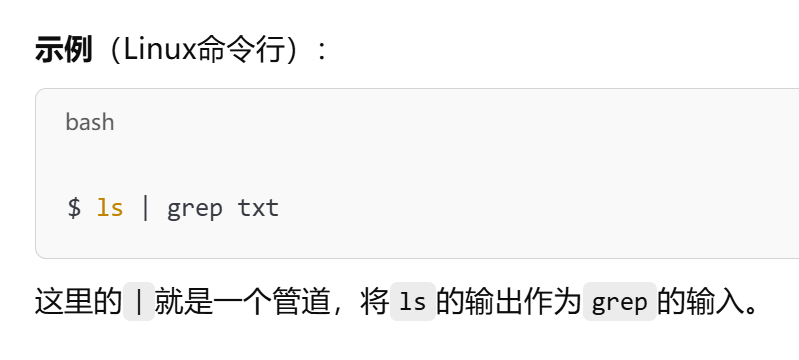
当访问的虚拟地址不在物理内存中时，触发缺页中断，将数据从硬盘调入内存

分页按固定大小划分，分段按逻辑划分

进程间通信（Inter-Process Communication, IPC）

1、管道

以字节流半双工通信，阻塞同步，通常用于父子进程**之间的通信**



2、命名管道（FIFO）

**具有**名字**标识**，支持多对多通信，通常用于同一主机的、无亲缘关系的进程通信



3、消息队列

相比命名管道（FIFO），支持随机访问、支持消息持久化。

通常用于日志系统

4、共享内存

速度最快，因为数据不需要在内核空间和用户空间之间来回拷贝。

需要同步机制（信号量、互斥锁）

**适用于高性能数据传输**的视频流

5、信号量

实现同步和互斥，**解决**死锁**问题**

**适用于**数据库连接池等。

6、信号

用于向进程发送异步通知或中断，固定编号、数量有限、可能丢失。



7、套接字

支持分布式通信、不同主机进程之间进行通信，使用TCP UDP网络协议传输

适用于网络通信

多路复用技术（Multiplexing）

通过单线程/进程同时监视多个I/O事件，用于

1. 网络服务器监视多个客户端连接
2. 文件系统监听多个文件的读写状态
3. 管理多个数据库连接

select()

通过一个数组监视多个文件描述符，检查哪些可读/可写   
数组大小限制 文件描述符上限1024、整体扫描 效率低下

poll()

改进用链表监视文件描述符  
解决了大小限制、但整体遍历 效率低下

epoll() Linux 提供

epoll\_create 创建一个 epoll 实例：

1. epoll\_ctl 增加、修改或删除要控制的文件描述符。
2. epoll\_wait 则是用于等待事件的发生。

与 select 不同的是，epoll 每次只需传递发生的事件，不需要传递所有文件描述符，极大提高了效率。

* 水平触发（默认）：只要发现事件有未处理的数据就会再次通知
* 边缘触发：只会在状态变化（例如从无数据到有数据）时通知一次，适用于高性能场景

知识框架

线程和进程区别

**进程**：

是资源分配的基本单位，一个进程可以包含多个线程

进程有独立的上下文（代码、数据、文件）等，创建和切换开销大

进程之间是相互独立的，需要进程间通信（IPC）机制，如管道、消息队列、共享内存、套接字等

线程：

是CPU 调度的基本单位，属于进程

线程组内共享资源，但有独立的PC和堆栈，创建和切换开销较小

线程直接读写内存即可，但需要同步机制以避免数据竞争

线程和协程的区别与联系

线程： 是操作系统调度的最小执行单元

由操作系统抢占式调度，可以多线程并行执行

更适用于 CPU 密集型任务

协程： 是在用户空间调度的最小执行单元

由程序员显式地通过非抢占式调度实现协作式并发，一般在单个线程内执行

更适用于 I/O 密集型任务