

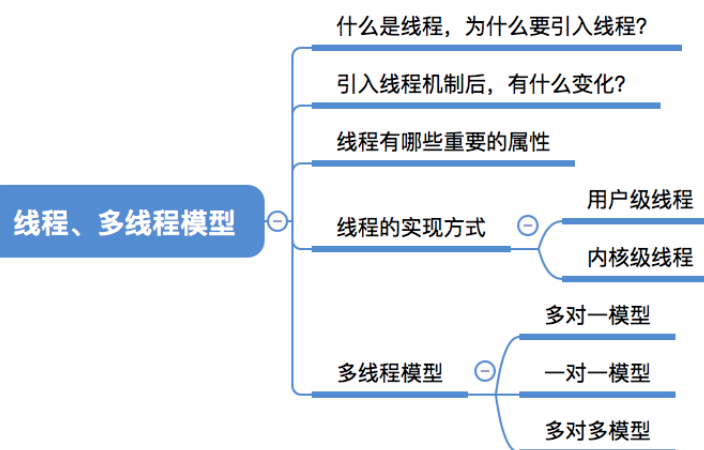
本节内容

线程概念 多线程模型

王道考研/CSKAOYAN.COM

1

知识总览



王道考研/CSKAOYAN.COM

2

什么是线程，为什么要引入线程？

故事发生在很久以前……

还没引入进程之前，系统中各个程序只能串行执行。

引入了进程之后…

But...Think think QQ 可以 do what?

进程是程序的一次执行。但这些功能显然不可能是一个程序顺序处理就能实现的

王道考研/CSKAOYAN.COM

3

什么是线程，为什么要引入线程？

有的进程可能需要“同时”做很多事，而传统的进程只能串行地执行一系列程序。为此，引入了“线程”，来增加并发度。

传统的进程是程序执行流的最小单位

引入线程后，线程成为了程序执行流的最小单位

QQ视频聊天处理程序

QQ传送文件处理程序

代码1 代码a 代码2 代码b 代码3 代码c 进程1 进程2 CPU

代码1 代码1 代码2 代码2 代码3 代码3 线程1 线程2 进程

代码5 代码d 代码6 代码e 代码7 代码f 线程3 线程4 进程

王道考研/CSKAOYAN.COM

4

什么是线程，为什么要引入线程？

可以把线程理解为“轻量级进程”。

线程是一个基本的CPU执行单元，也是程序执行流的最小单位。

引入线程之后，不仅是进程之间可以并发，进程内的各线程之间也可以并发，从而进一步提升了系统的并发度，使得一个进程内也可以并发处理各种任务（如QQ视频、文字聊天、传文件）

引入线程后，进程只作为除CPU之外的系统资源的分配单元（如打印机、内存地址空间等都是分配给进程的）。

线程则作为处理机的分配单元。

王道考研/CSKAOYAN.COM

5

引入线程机制后，有什么变化？

带来的变化

- 资源分配、调度**
 - 传统进程机制中，进程是资源分配、调度的基本单位
 - 引入线程后，进程是资源分配的基本单位，线程是调度的基本单位
- 并发性**
 - 传统进程机制中，只能进程间并发
 - 引入线程后，各线程间也能并发，提升了并发度
- 系统开销**
 - 传统的进程间并发，需要切换进程的运行环境，系统开销很大
 - 线程间并发，如果是同一进程内的线程切换，则不需要切换进程环境，系统开销小
 - 引入线程后，并发所带来的系统开销减小

类比：
去图书馆看书。桌子=处理机，人=进程，看不同的书=线程
切换进程运行环境：有一个不认识的人要用桌子，你需要把你的书收走，他把自己的书放到桌上
同一进程内的线程切换=你的舍友要用这张书桌，可以不把桌子上的书收走

王道考研/CSKAOYAN.COM

6

