

本节内容

基本分段存储管理方式

王道考研/CSKAOYAN.COM

知识总览

基本分段存储管理

与“分页”最大的区别就是——离散分配时所分配地址空间的基本单位不同

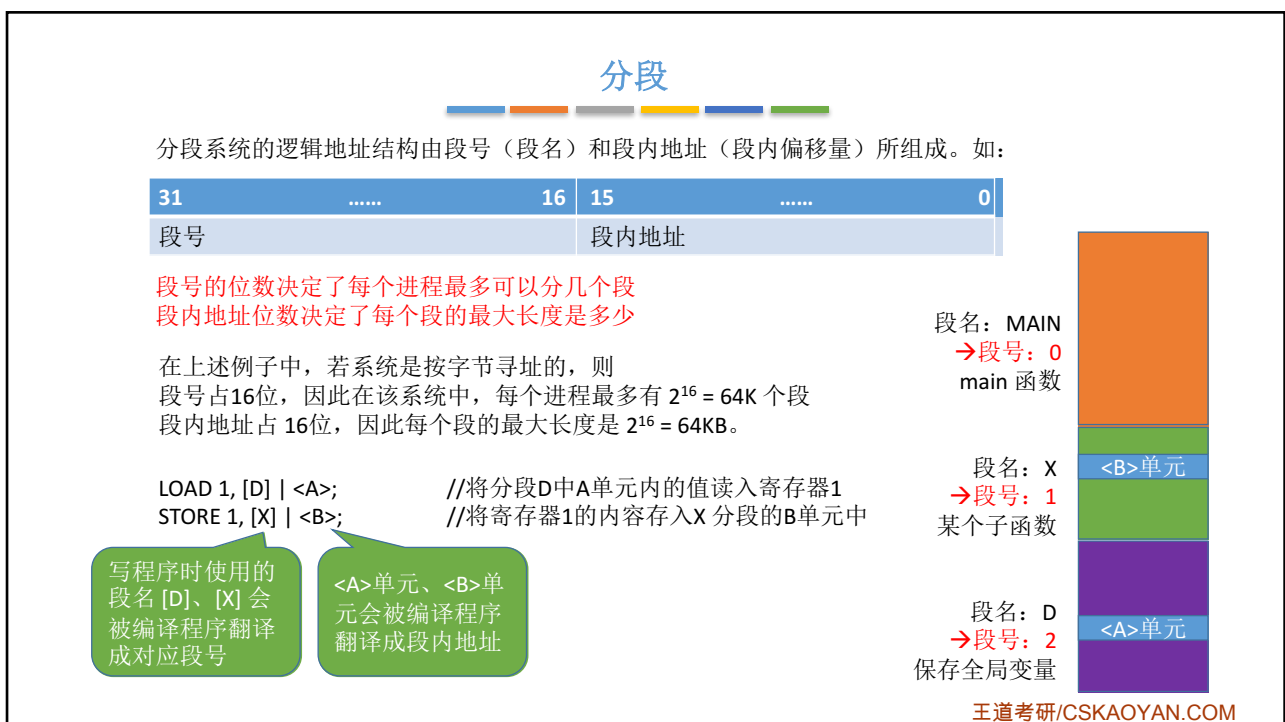
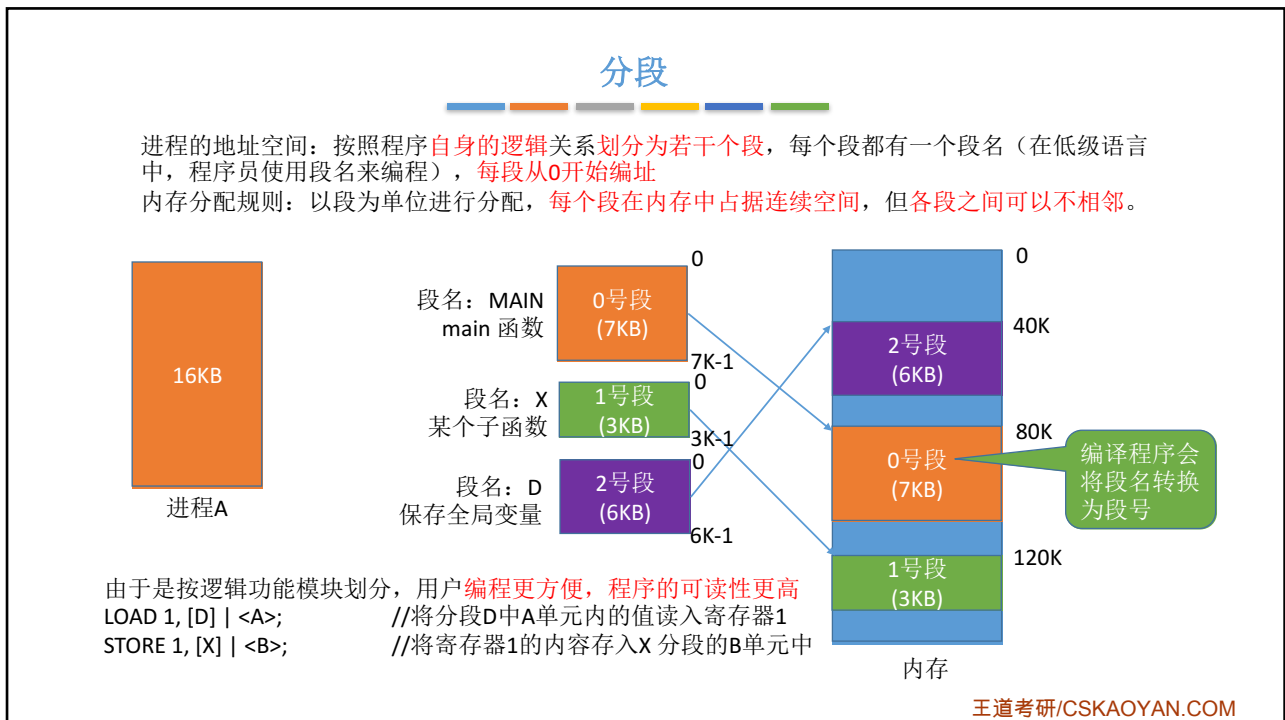
什么是分段（类似于分页管理中的“分页”）

什么是段表（类似于分页管理中的“页表”）

如何实现地址变换

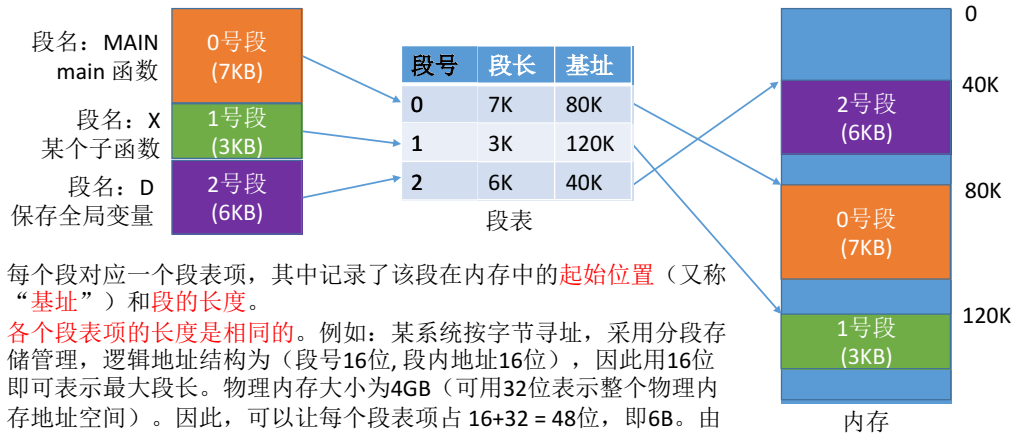
分段、分页管理的对比

王道考研/CSKAOYAN.COM



段表

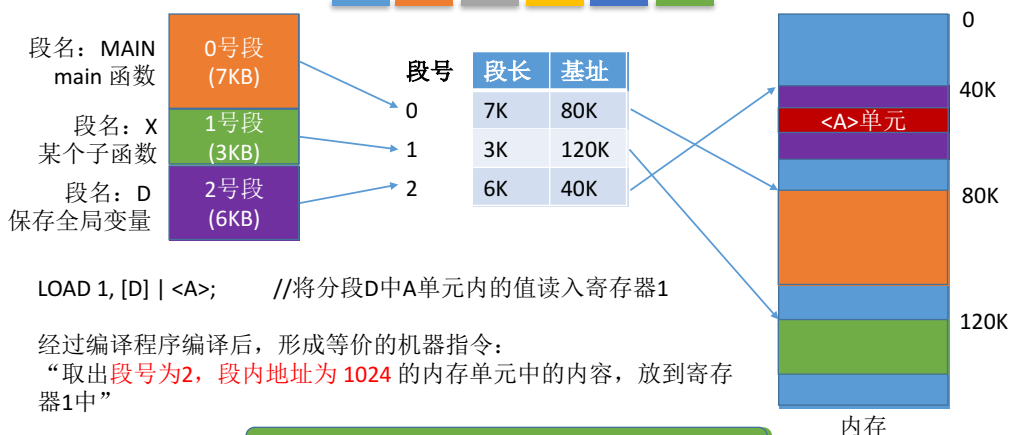
问题: 程序分多个段, 各段离散地装入内存, 为了保证程序能正常运行, 就必须能从物理内存中找到各个逻辑段的存放位置。为此, 需为每个进程建立一张段映射表, 简称“**段表**”。



1. 每个段对应一个段表项, 其中记录了该段在内存中的**起始位置**(又称“**基址**”)和**段的长度**。
2. **各个段表项的长度是相同的**。例如: 某系统按字节寻址, 采用分段存储管理, 逻辑地址结构为(段号16位, 段内地址16位), 因此用16位即可表示最大段长。物理内存大小为4GB(可用32位表示整个物理内存地址空间)。因此, 可以让每个段表项占 $16+32=48$ 位, 即6B。由于段表项长度相同, 因此**段号可以是隐含的, 不占存储空间**。若段表存放的起始地址为 M , 则 K 号段对应的段表项存放的地址为 $M + K*6$

王道考研/CSKAOYAN.COM

地址变换



LOAD 1, [D] | <A>; //将分段D中A单元内的值读入寄存器1

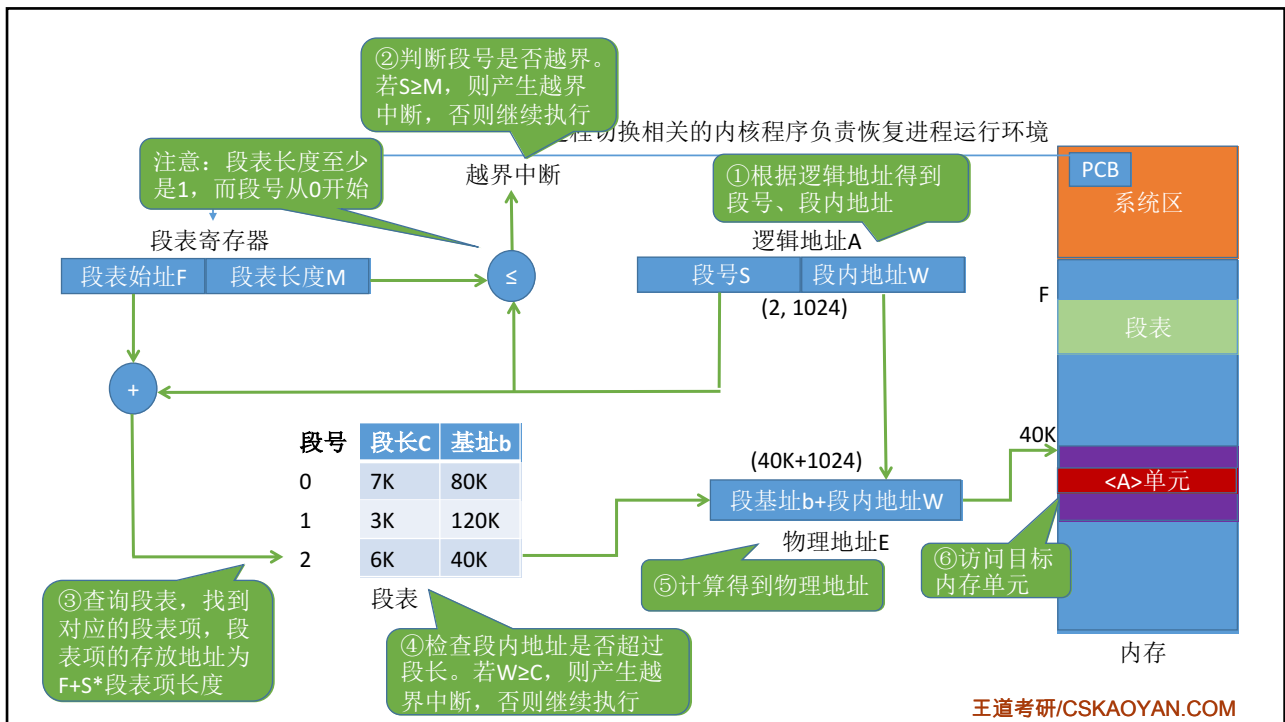
经过编译程序编译后, 形成等价的机器指令:

“取出**段号为2**, **段内地址为1024**的内存单元中的内容, 放到寄存器1中”

CPU执行指令时需要将逻辑地址变换为物理地址

机器指令中的逻辑地址用二进制表示: 00000000000000100000000100000000

王道考研/CSKAOYAN.COM



分段、分页管理的对比

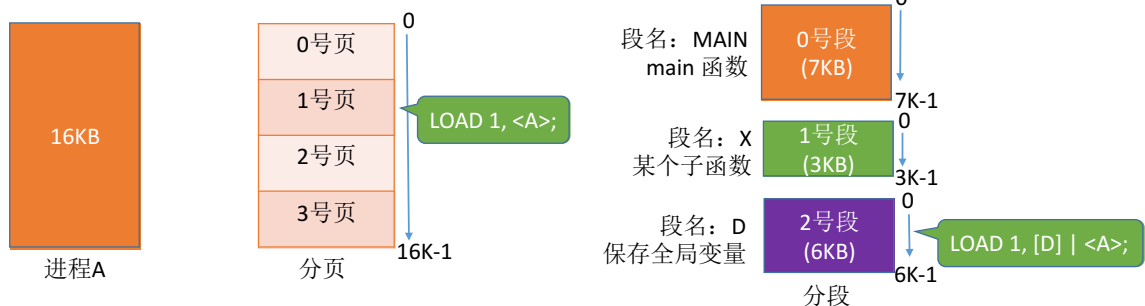
页是信息的物理单位。分页的主要目的是为了实现离散分配, 提高内存利用率。分页仅仅是系统管理上的需要, 完全是系统行为, 对用户是不可见的。

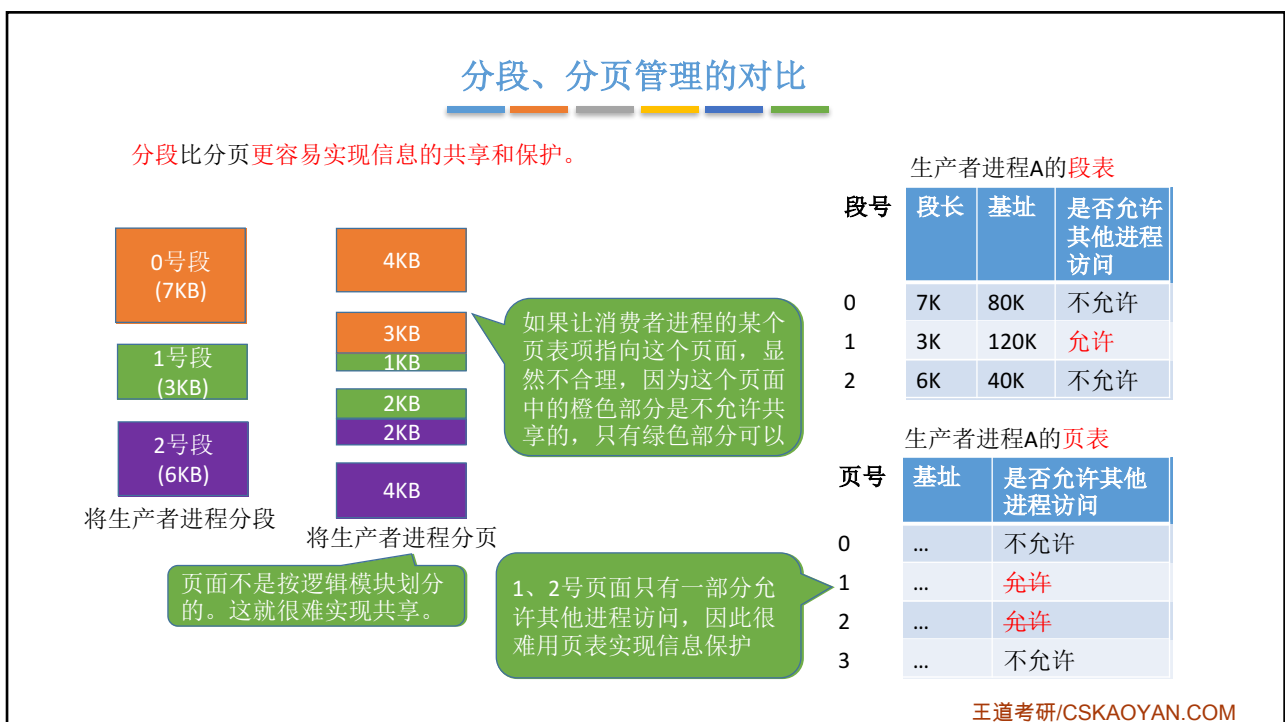
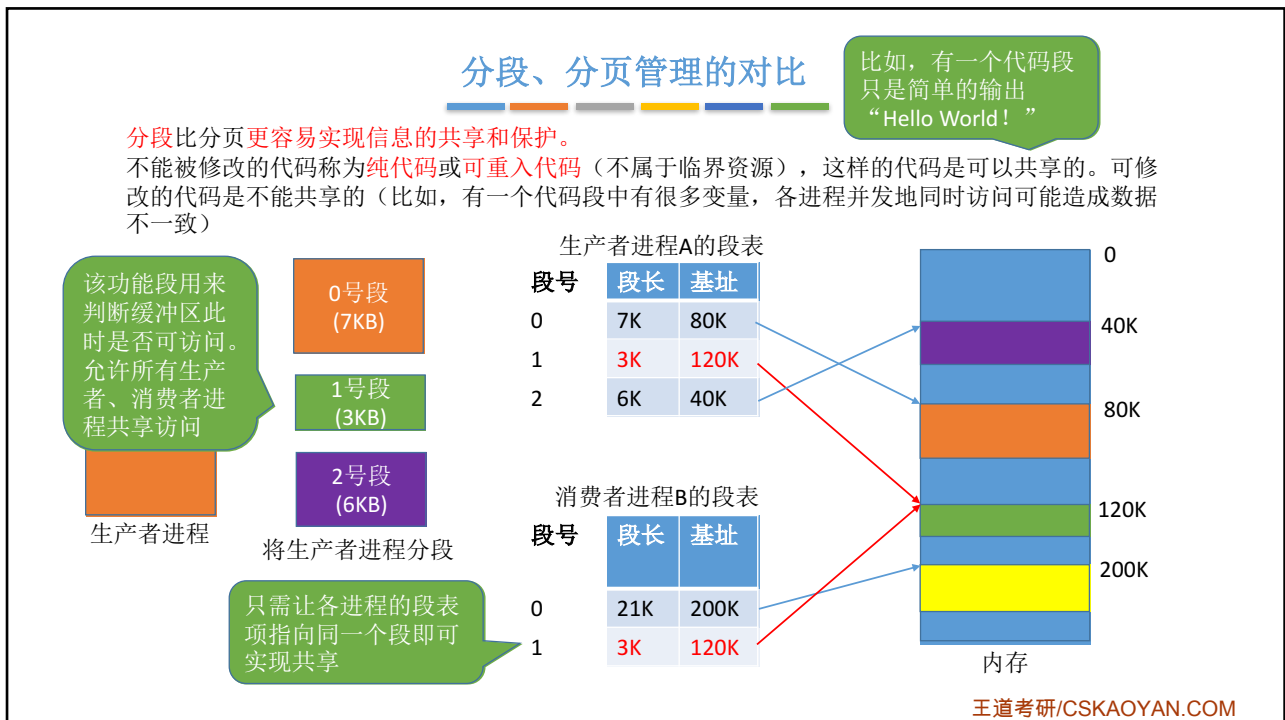
段是信息的逻辑单位。分段的主要目的是更好地满足用户需求。一个段通常包含着一组属于一个逻辑模块的信息。分段对用户是可见的, 用户编程时需要显式地给出段名。

页的大小固定且由系统决定。段的长度却不固定, 决定于用户编写的程序。

分页的用户进程地址空间是一维的, 程序员只需给出一个记忆符即可表示一个地址。

分段的用户进程地址空间是二维的, 程序员在标识一个地址时, 既要给出段名, 也要给出段内地址。





分段、分页管理的对比

页是信息的物理单位。分页的主要目的是为了实现离散分配，提高内存利用率。分页仅仅是系统管理上的需要，完全是系统行为，**对用户是不可见的**。

段是信息的逻辑单位。分页的主要目的是更好地满足用户需求。一个段通常包含着一组属于一个逻辑模块的信息。**分段对用户是可见的**，用户编程时需要显式地给出段名。

页的大小固定且由系统决定。段的长度却不固定，决定于用户编写的程序。

分页的用户进程地址空间是一维的，程序员只需给出一个记忆符即可表示一个地址。

分段的用户进程地址空间是二维的，程序员在标识一个地址时，既要给出段名，也要给出段内地址。

分段比分页更容易实现信息的共享和保护。不能被修改的代码称为**纯代码**或**可重入代码**（不属于临界资源），这样的代码是可以共享的。可修改的代码是不能共享的

访问一个逻辑地址需要几次访存？

分页（单级页表）：第一次访存——查内存中的页表，第二次访存——访问目标内存单元。总共**两次访存**

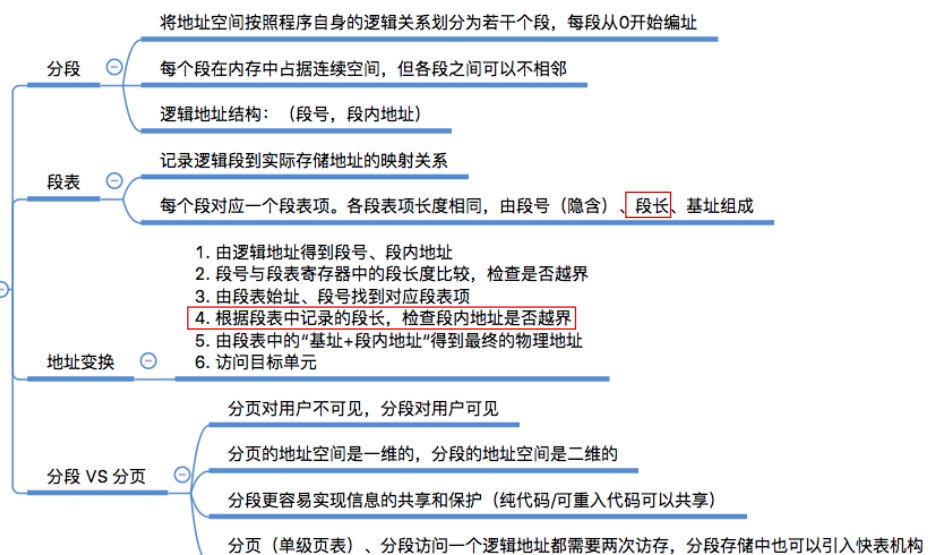
分段：第一次访存——查内存中的段表，第二次访存——访问目标内存单元。总共**两次访存**

与分页系统类似，分段系统中也可以引入**快表机构**，将近期访问过的段表项放到快表中，这样可以**少一次访问**，加快地址变换速度。

王道考研/CSKAOYAN.COM

知识回顾与重要考点

基本分段存储管理



王道考研/CSKAOYAN.COM