

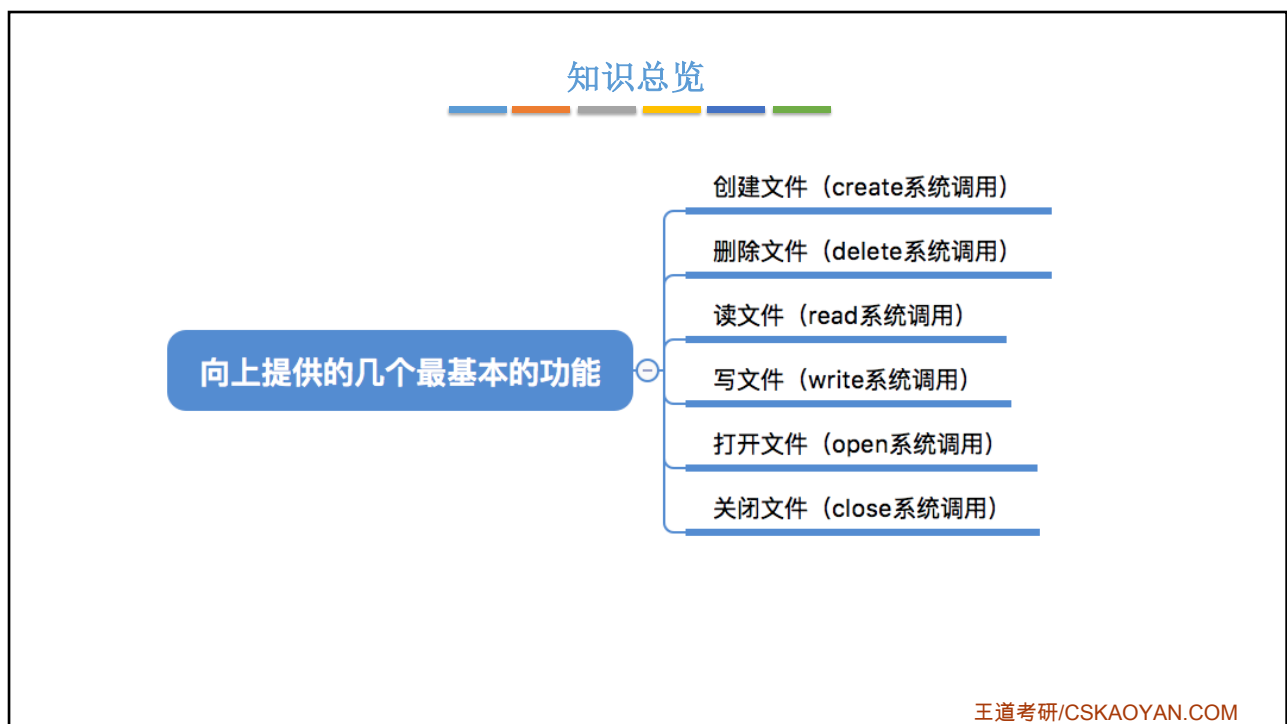
2020/5/20

本节内容

文件的基本操作

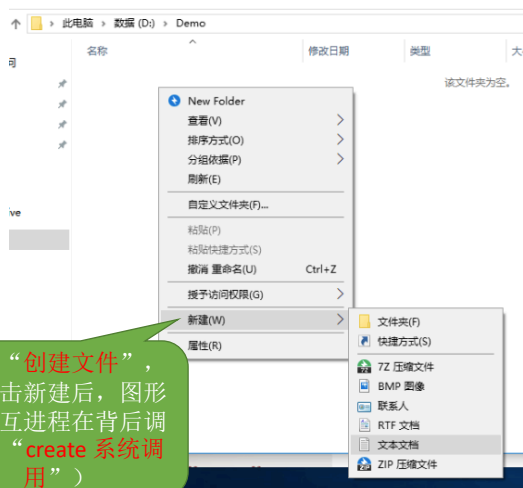
王道考研/CSKAOYAN.COM

1



2

创建文件



进行 Create 系统调用时，需要提供的几个主要参数：

1. 所需的外存空间大小（如：一个盘块，即1KB）
2. 文件存放路径（“D:/Demo”）
3. 文件名（这个地方默认为“新建文本文档.txt”）

操作系统在处理 Create 系统调用时，主要做了两件事：

1. 在外存中找到文件所需的**空间**（结合上小节学习的空闲链表法、位示图、成组链接法等管理策略，找到空闲空间）
2. 根据文件存放路径的信息找到该目录对应的目录文件（此处就是 D:/Demo 目录），在目录中**创建该文件对应的目录项**。目录项中包含了文件名、文件在外存中的存放位置等信息。

王道考研/CSKAOYAN.COM

3

删除文件



进行 Delete 系统调用时，需要提供的几个主要参数：

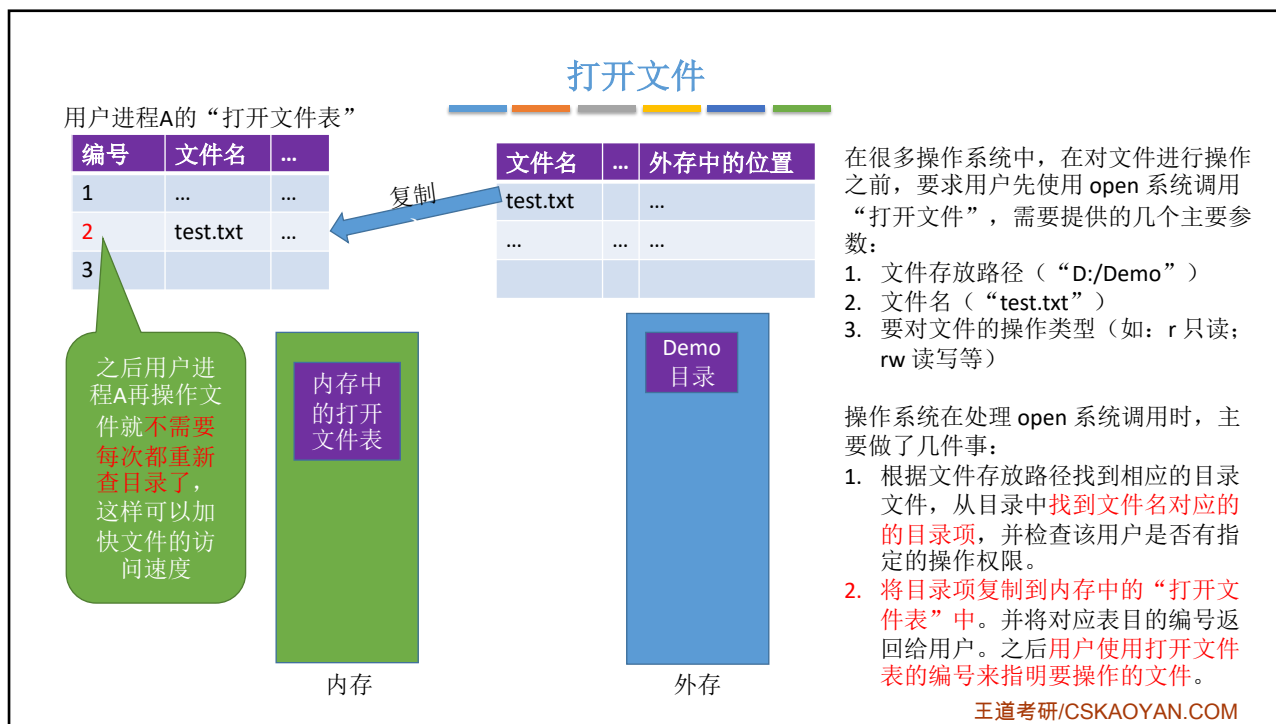
1. 文件存放路径（“D:/Demo”）
2. 文件名（“test.txt”）

操作系统在处理 Delete 系统调用时，主要做了几件事：

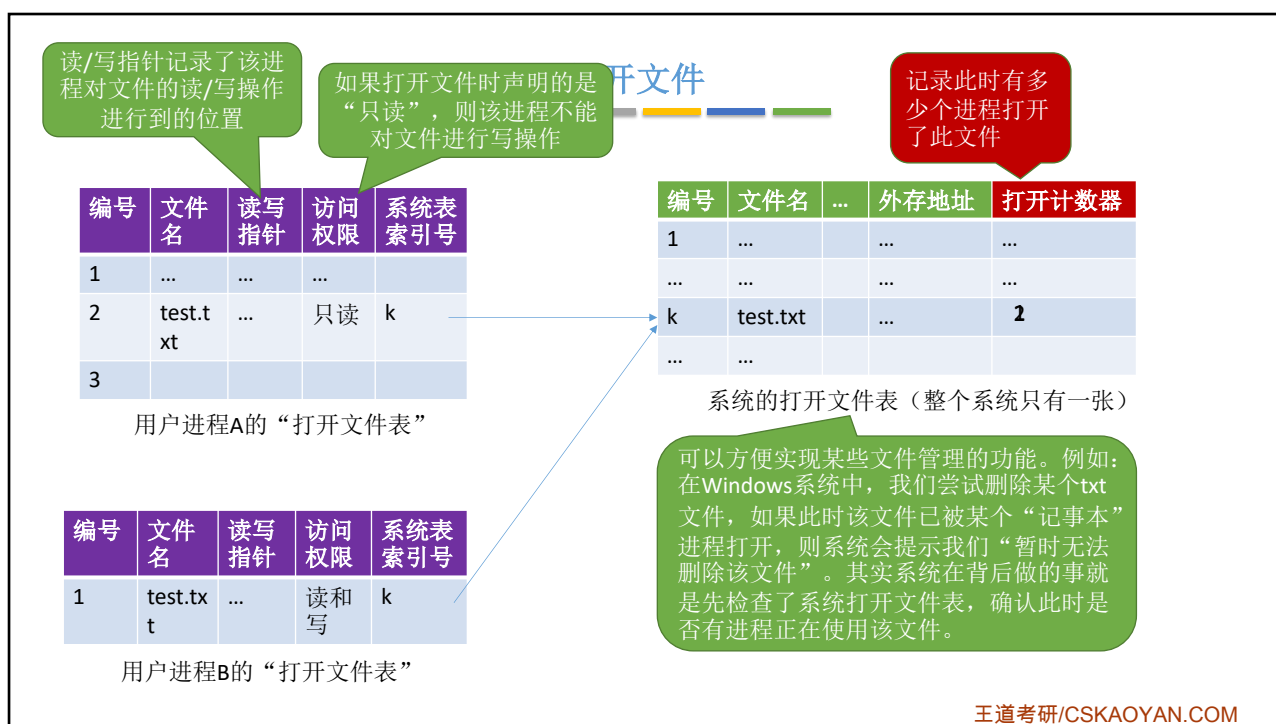
1. 根据文件存放路径找到相应的目录文件，从目录中**找到文件名对应的目录项**。
2. 根据该目录项记录的文件在外存的存放位置、文件大小等信息，**回收文件占用的磁盘块**。（回收磁盘块时，根据空闲表法、空闲链表法、位图法等管理策略的不同，需要做不同的处理）
3. 从目录表中**删除文件对应的目录项**。

王道考研/CSKAOYAN.COM

4



5



6

2020/5/20

关闭文件

记录此时有多少个进程打开了此文件

| 编号 | 文件名 | 读写指针 | 访问权限 | 系统表索引号 |
|----|----------|------|------|--------|
| 1 | ... | ... | ... | ... |
| 2 | test.txt | ... | 只读 | k |
| 3 | ... | ... | ... | ... |

用户进程A的“打开文件表”

| 编号 | 文件名 | ... | 外存地址 | 打开计数器 |
|-----|----------|-----|------|-------|
| 1 | ... | ... | ... | ... |
| ... | ... | ... | ... | ... |
| k | test.txt | ... | ... | 2 |
| ... | ... | ... | ... | ... |

系统的打开文件表（整个系统只有一张）

| 编号 | 文件名 | 读写指针 | 访问权限 | 系统表索引号 |
|----|-----|------|------|--------|
|----|-----|------|------|--------|

用户进程B的“打开文件表”

进程使用完文件后，要“关闭文件”

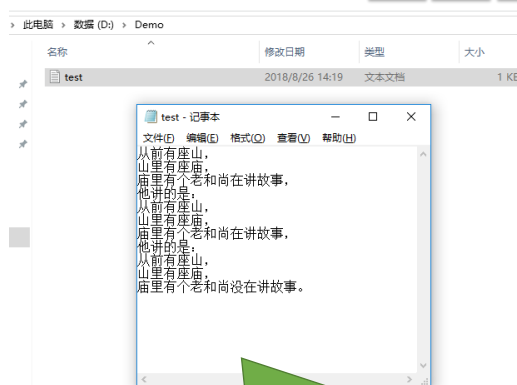
操作系统在处理 Close 系统调用时，主要做了几件事：

1. 将进程的打开文件表相应表项删除
2. 回收分配给该文件的内存空间等资源
3. 系统打开文件表的打开计数器count 减1，若 count = 0，则删除对应表项。

王道考研/CSKAOYAN.COM

7

读文件



可以“读文件”，将文件数据读入内存，才能让CPU处理（双击后，“记事本”应用程序通过操作系统提供的“读文件”功能，即 **read 系统调用**，将文件数据从外存读入内存，并显示在屏幕上）

| 编号 | 文件名 | 读写指针 | 访问权限 | ... |
|----|----------|------|------|-----|
| 1 | test.txt | ... | 读和写 | ... |

“记事本”进程的“打开文件表”

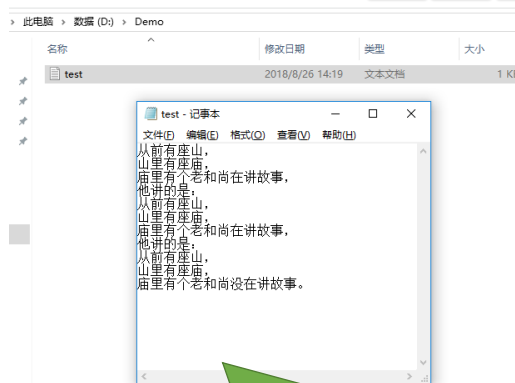
进程使用 read 系统调用完成读操作。需要指明是哪个文件（在支持“打开文件”操作的系统中，只需要提供文件在打开文件表中的索引号即可），还需要指明要读入多少数据（如：读入 1KB）、指明读入的数据要放在内存中的什么位置。

操作系统在处理 read 系统调用时，会从读指针指向的外存中，将用户指定大小的数据读入用户指定的内存区域中。

王道考研/CSKAOYAN.COM

8

写文件



| 编号 | 文件名 | 读写指针 | 访问权限 | ... |
|----|----------|------|------|-----|
| 1 | test.txt | ... | 读和写 | ... |

“记事本”进程的“打开文件表”

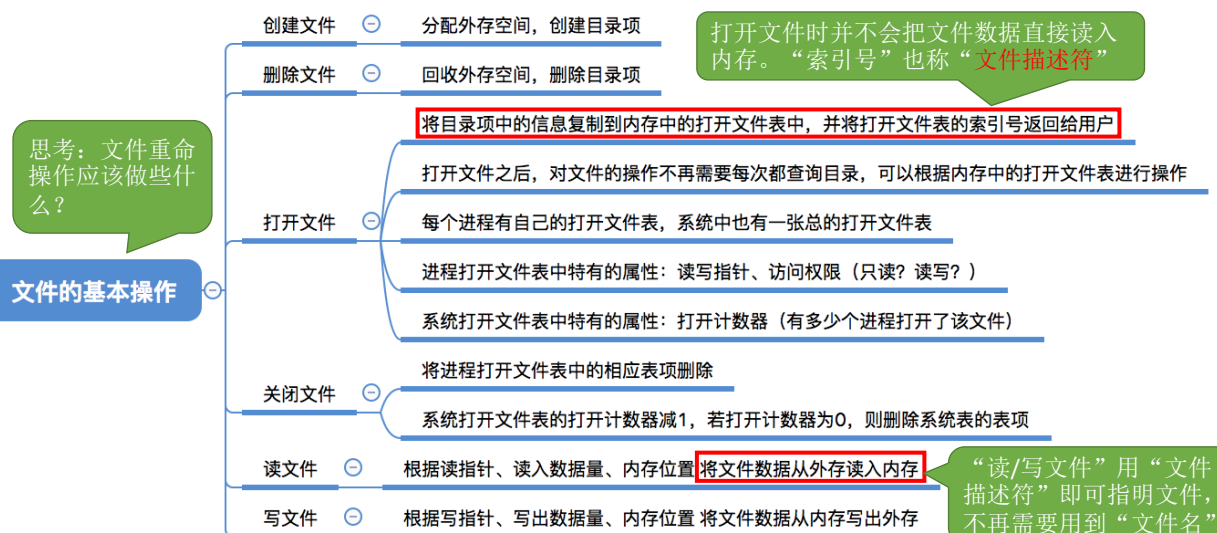
可以“写文件”，将更改过的文件数据写回外存（我们在“记事本”应用程序中编辑文件内容，点击“保存”后，“记事本”应用程序通过操作系统提供的“写文件”功能，即 **write 系统调用**，将文件数据从内存写回外存）

进程使用 **write** 系统调用完成写操作，需要指明是哪个文件（在支持“打开文件”操作的系统中，只需要提供文件在打开文件表中的索引号即可），还需要指明要写出多少数据（如：写出 **1KB**）、写回外存的数据放在内存中的什么位置
操作系统在处理 **write** 系统调用时，会从用户指定的内存区域中，将指定大小的数据写回写指针指向的外存。

王道考研/CSKAOYAN.COM

9

知识点回顾与重要考点



王道考研/CSKAOYAN.COM

10