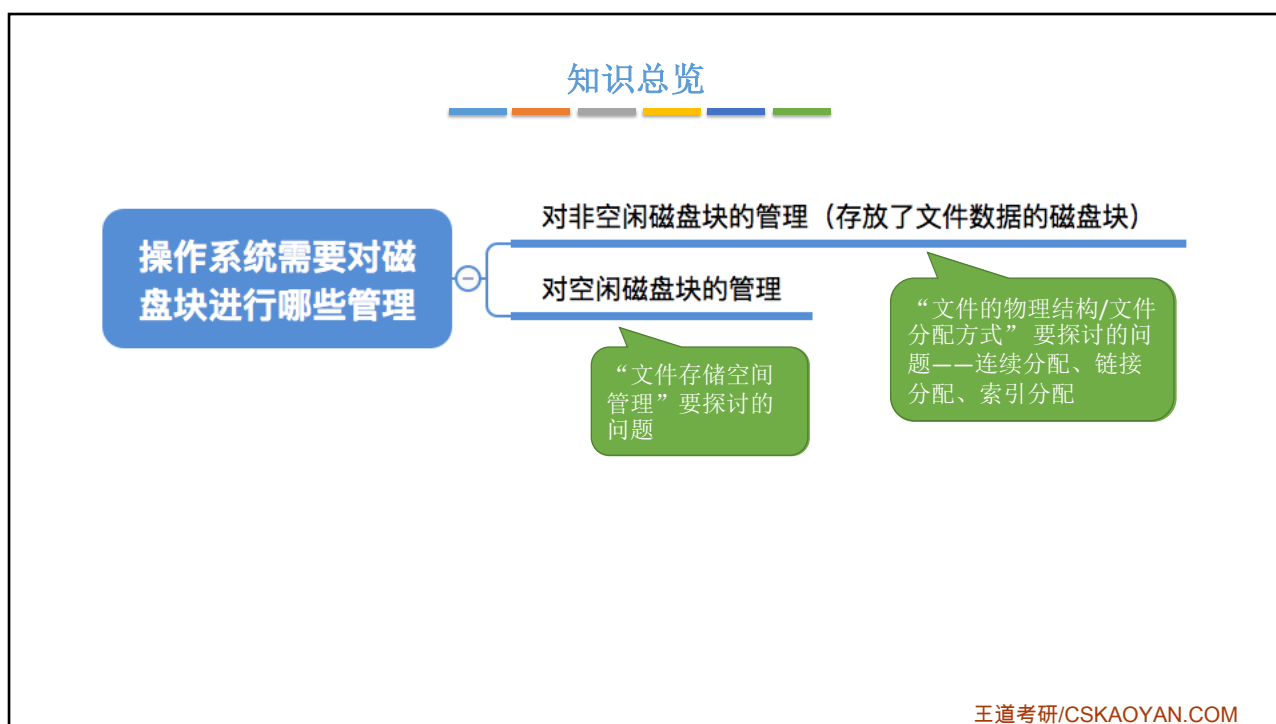


本节内容

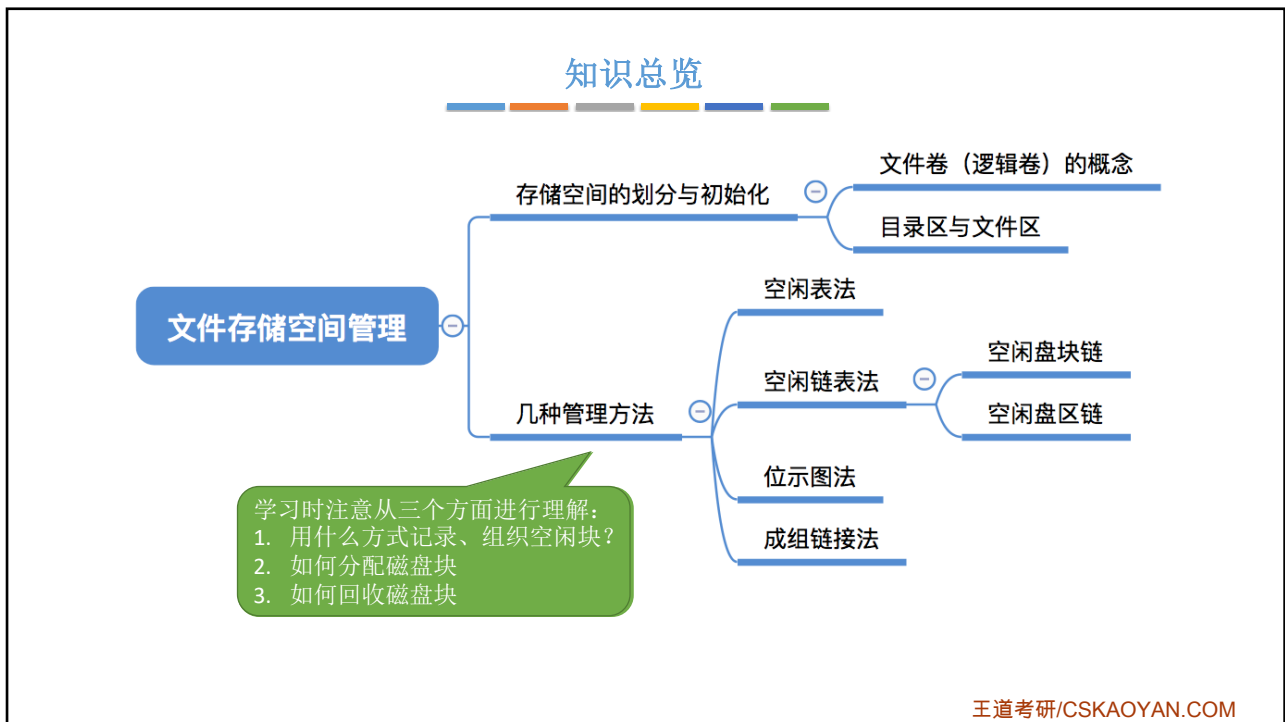
文件存储空间管理

王道考研/CSKAOYAN.COM

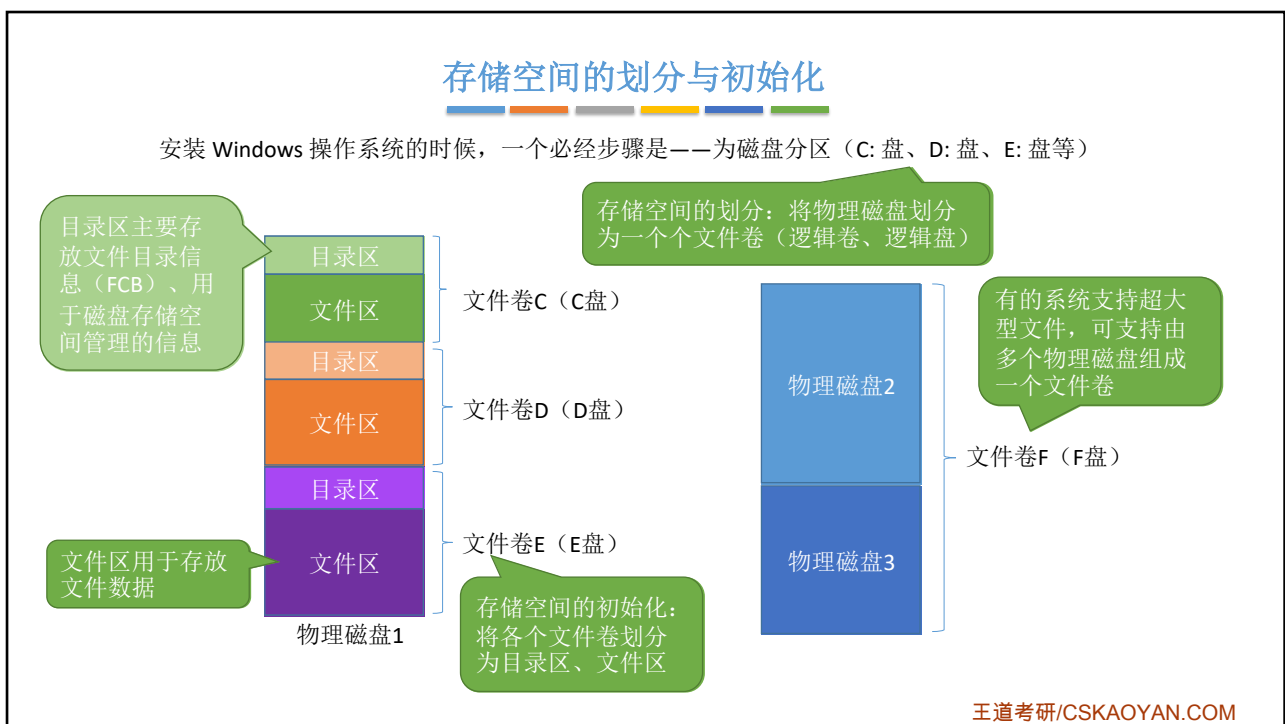
1



2



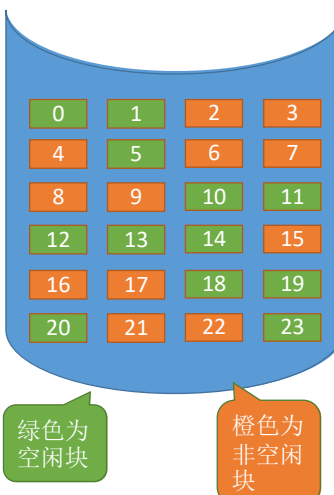
3



4

存储空间管理——空闲表法

适用于“连续分配方式”



第一个空闲盘块号	空闲盘块数
0	2
5	1
13	2
18	3
23	1

空闲盘块表

如何分配磁盘块：与内存管理中的动态分区分配很类似，为一个文件分配连续的存储空间。同样可采用首次适应、最佳适应、最坏适应等算法来决定要为文件分配哪个区间。

Eg: 新创建的文件请求3个块，采用首次适应算法

绿色为空闲块

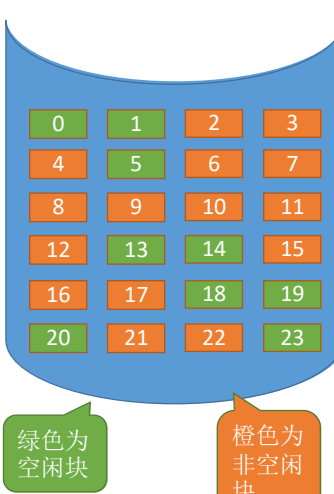
橙色为非空闲块

王道考研/CSKAOYAN.COM

5

存储空间管理——空闲表法

适用于“连续分配方式”



第一个空闲盘块号	空闲盘块数
0	2
5	1
13	8
23	1

空闲盘块表

如何分配磁盘块：与内存管理中的动态分区分配很类似，为一个文件分配连续的存储空间。同样可采用首次适应、最佳适应、最坏适应等算法来决定要为文件分配哪个区间。

如何回收磁盘块：与内存管理中的动态分区分配很类似，当回收某个存储区时需要四种情况——①回收区的前后都没有相邻空闲区；②回收区的前后都是空闲区；③回收区前面是空闲区；④回收区后面是空闲区。总之，回收时需要注意表项的合并问题。

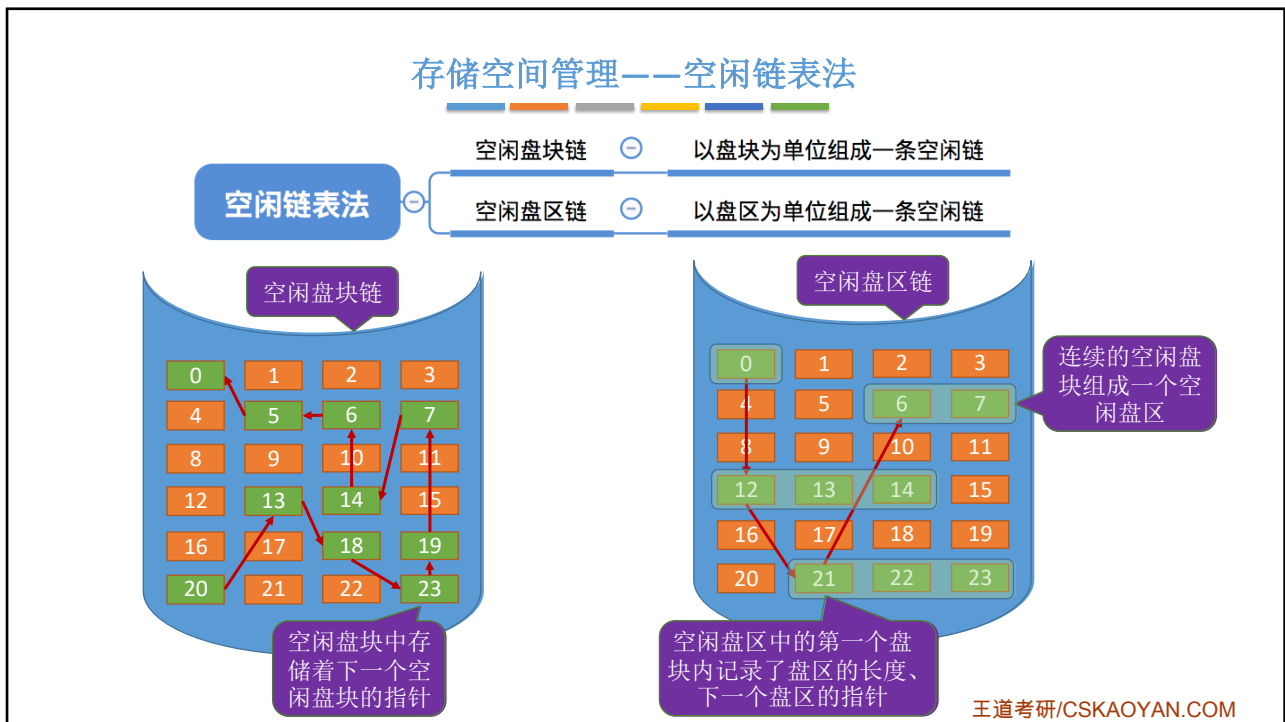
情况② Eg: 假设此时删除了某文件，系统回收了它占用的 15、16、17号块

绿色为空闲块

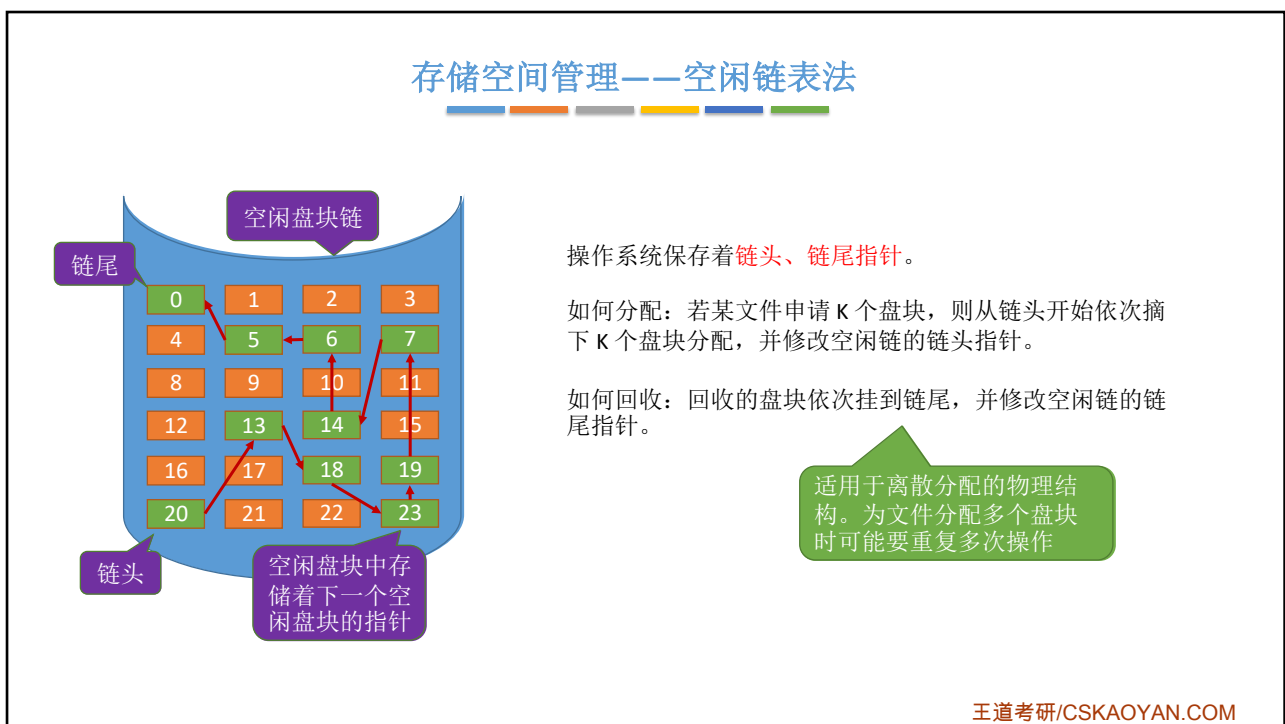
橙色为非空闲块

王道考研/CSKAOYAN.COM

6

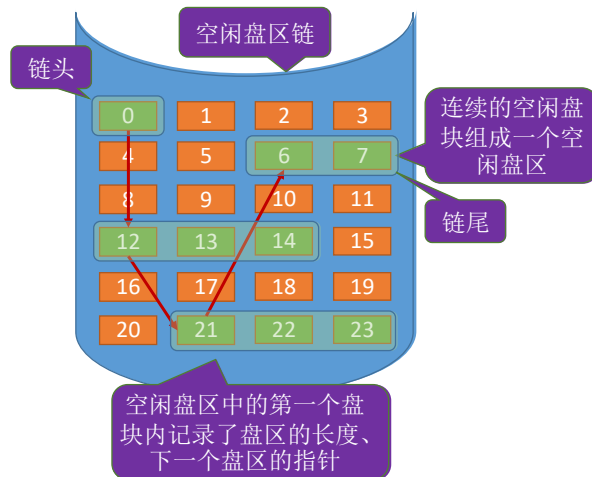


7



8

存储空间管理——空闲链表法



操作系统保存着链头、链尾指针。

如何分配：若某文件申请 K 个盘块，则可以采用首次适应、最佳适应等算法，从链头开始检索，按照算法规则找到一个大小符合要求的空闲盘区，分配给文件。若没有合适的连续空闲块，也可以将不同盘区的盘块同时分配给一个文件，注意分配后可能要修改相应的链指针、盘区大小等数据。

如何回收：若回收区和某个空闲盘区相邻，则需要将回收区合并到空闲盘区中。若回收区没有和任何空闲区相邻，将回收区作为单独的一个空闲盘区挂到链尾。

离散分配、连续分配都适用。为一个文件分配多个盘块时效率更高

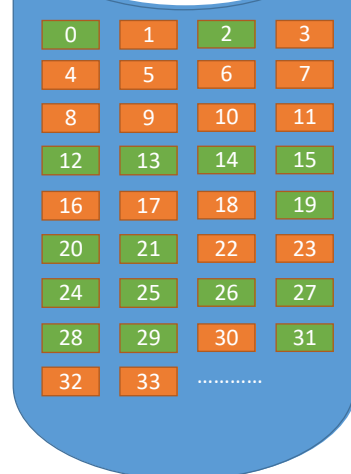
王道考研/CSKAOYAN.COM

9

存储空间管理——位示图法

$(0,1) \rightarrow b=16*0+1=1$
 $(1,10) \rightarrow b=16*1+10=26$

$b=13 \rightarrow i=13/16=0, j=13\%16=13$
 $b=31 \rightarrow i=31/16=1, j=31\%16=15$



字号	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	1	0	1	1	1	1	1	1	1	1	1	0	0	0	0
1	1	1	1	0	0	0	1	1	0	0	0	0	0	0	1	0
2	1	1	...													
...																

位示图：每个二进制位对应一个盘块。在本例中，“0”代表盘块空闲，“1”代表盘块已分配。位示图一般用连续的“字”来表示，如本例中一个字的字长是16位，字中的每一位对应一个盘块。因此可以用（字号，位号）对应一个盘块号。当然有的题目中也描述为（行号，列号）

重要重要重要：要能自己推出盘块号与（字号，位号）相互转换的公式。

注意题目条件：盘块号、字号、位号到底是从0开始还是从1开始
 如本例中盘块号、字号、位号从0开始，若 n 表示字长，则...

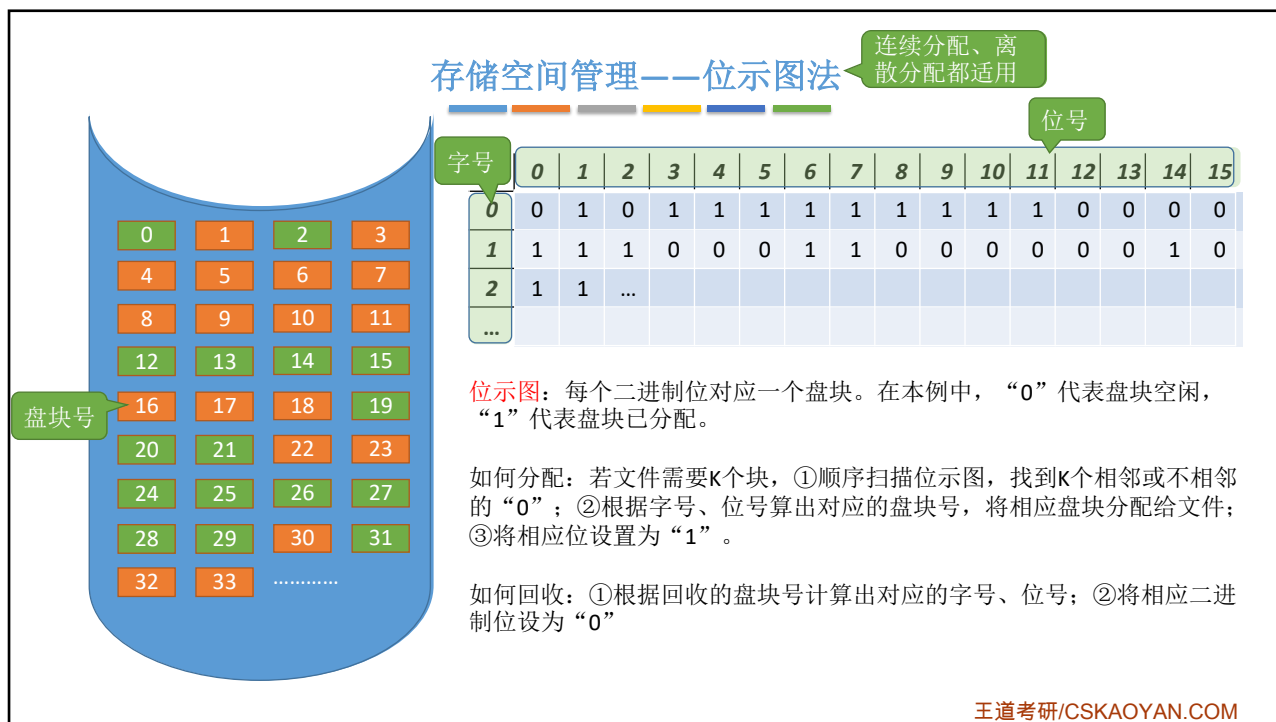
(字号, 位号)=(i, j) 的二进制位对应的 盘块号 $b = ni + j$

b 号盘块对应的字号 $i = b/n$ ，位号 $j = b\%n$

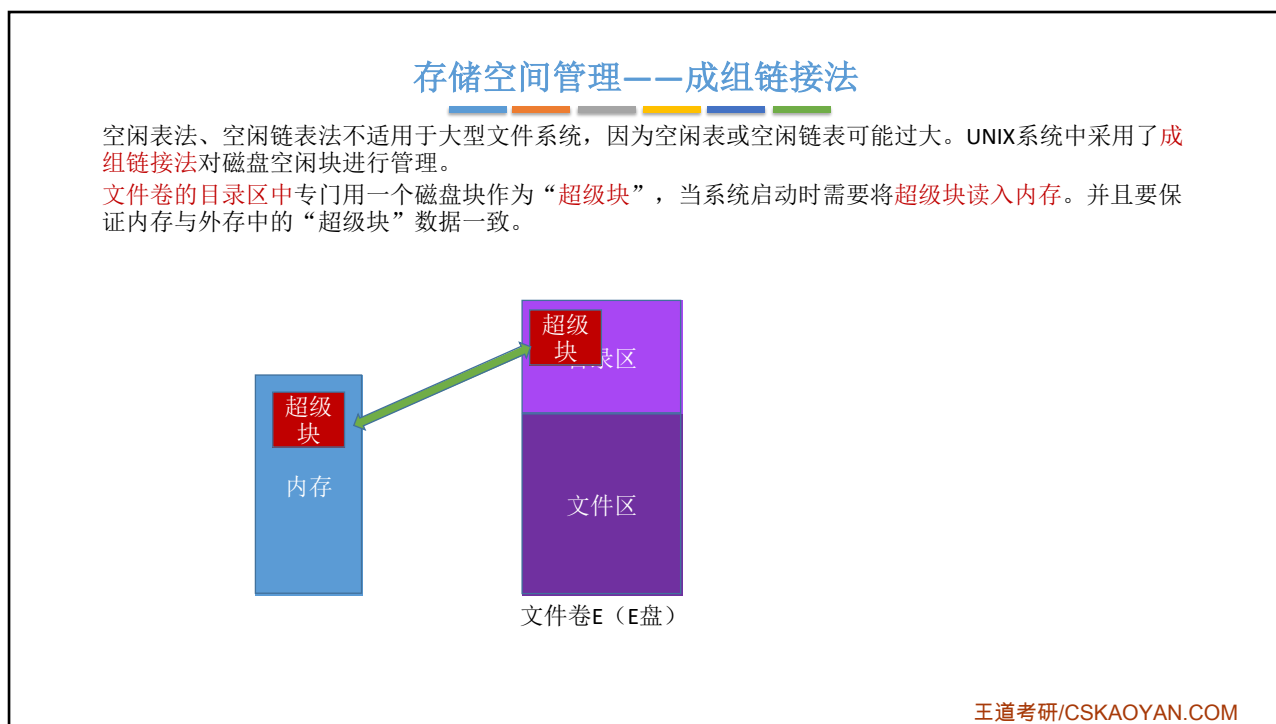
注：王道课本上给出的是盘块号、字号、位号都从1开始的算法。

王道考研/CSKAOYAN.COM

10

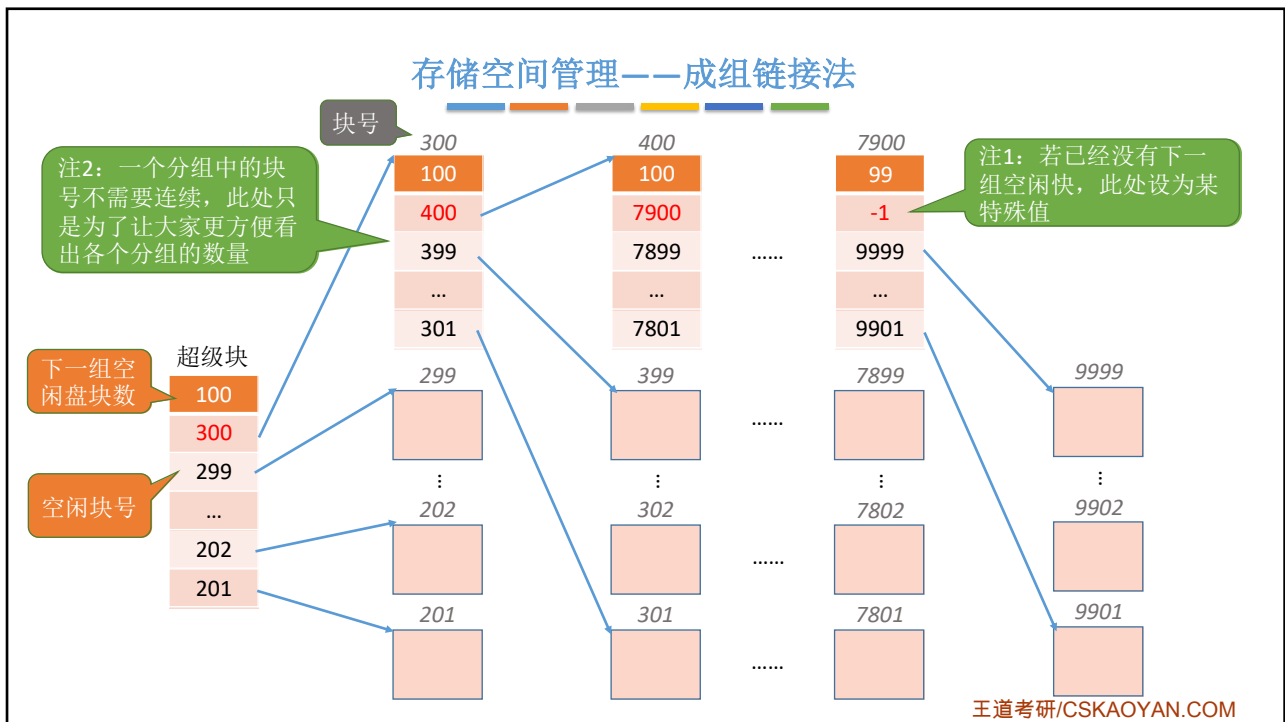


11

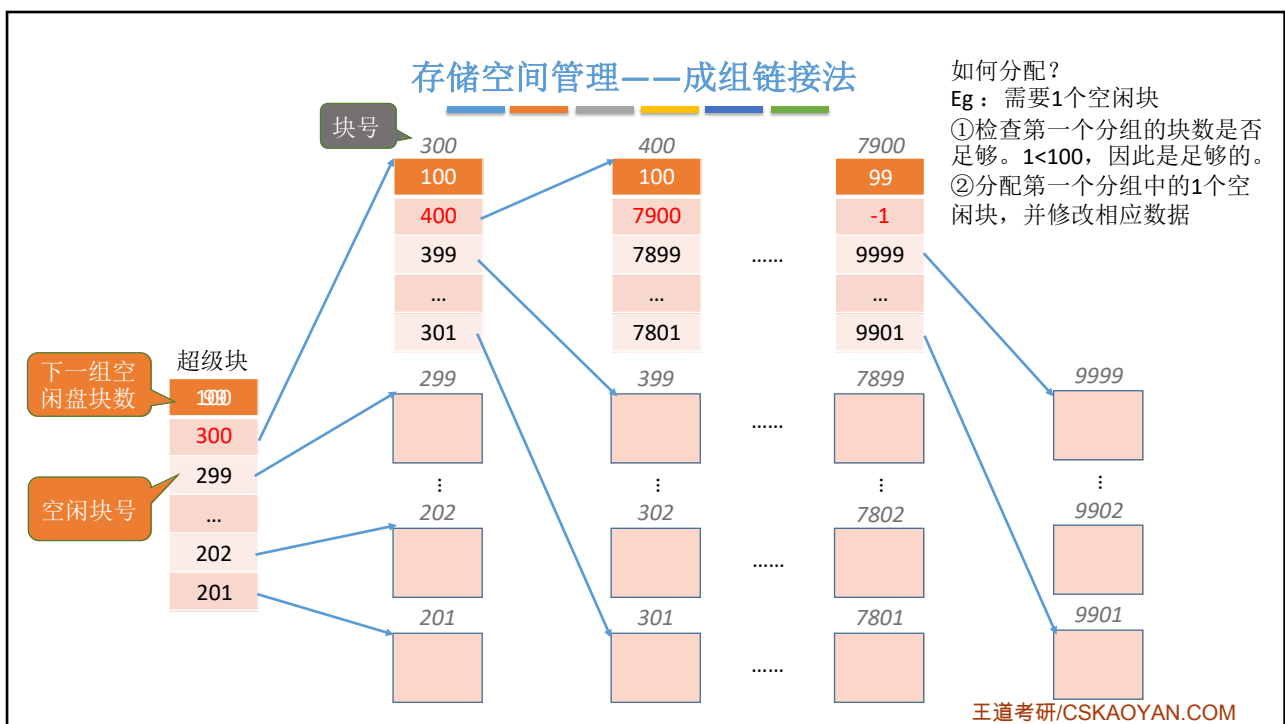


12

2020/5/20

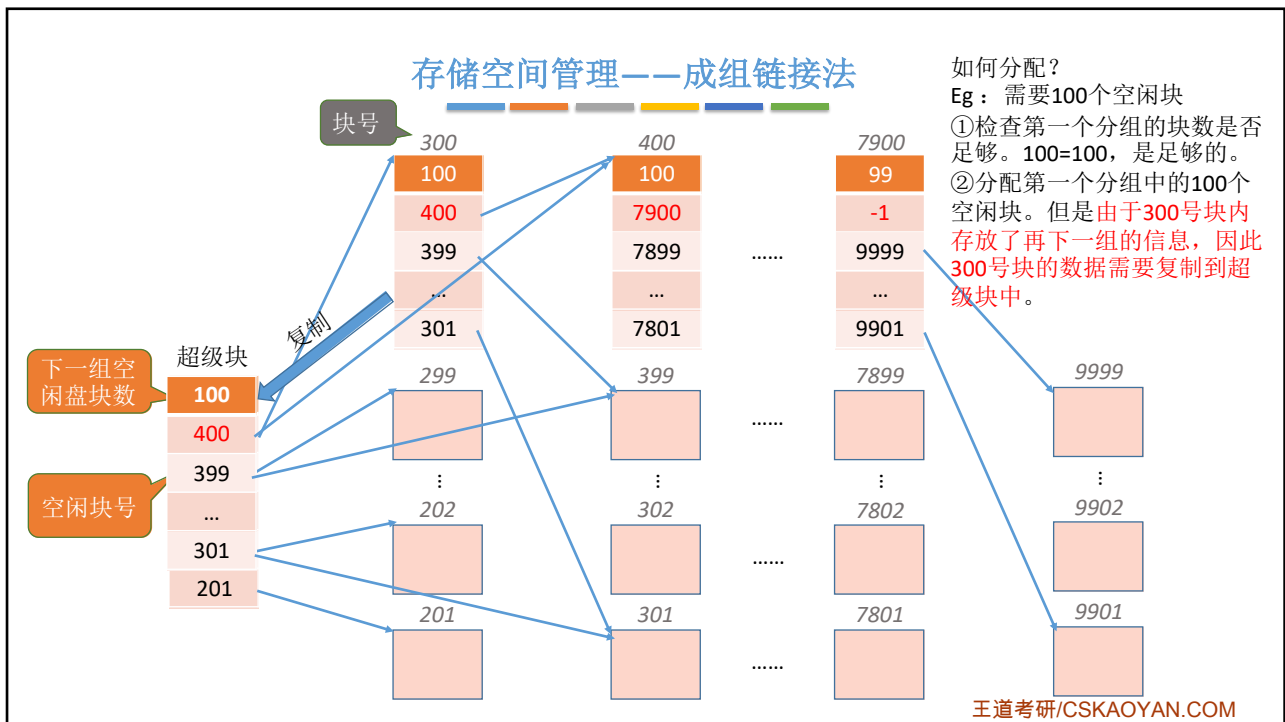


13

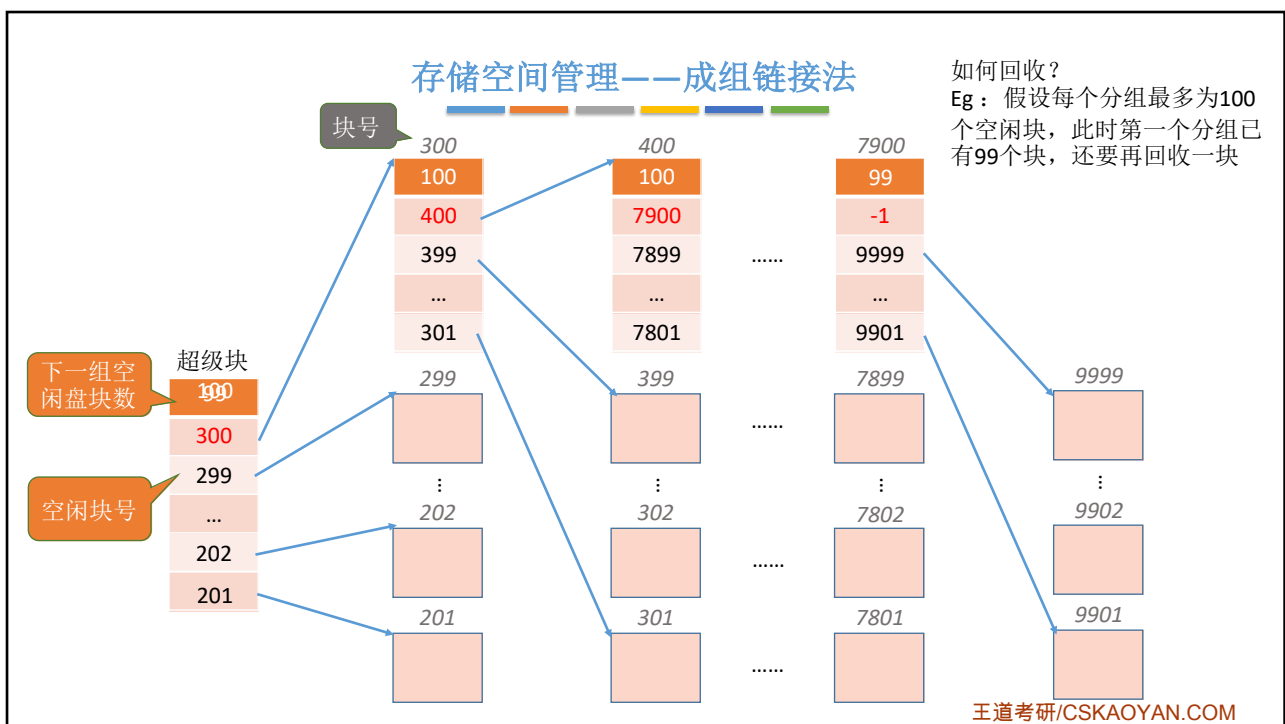


14

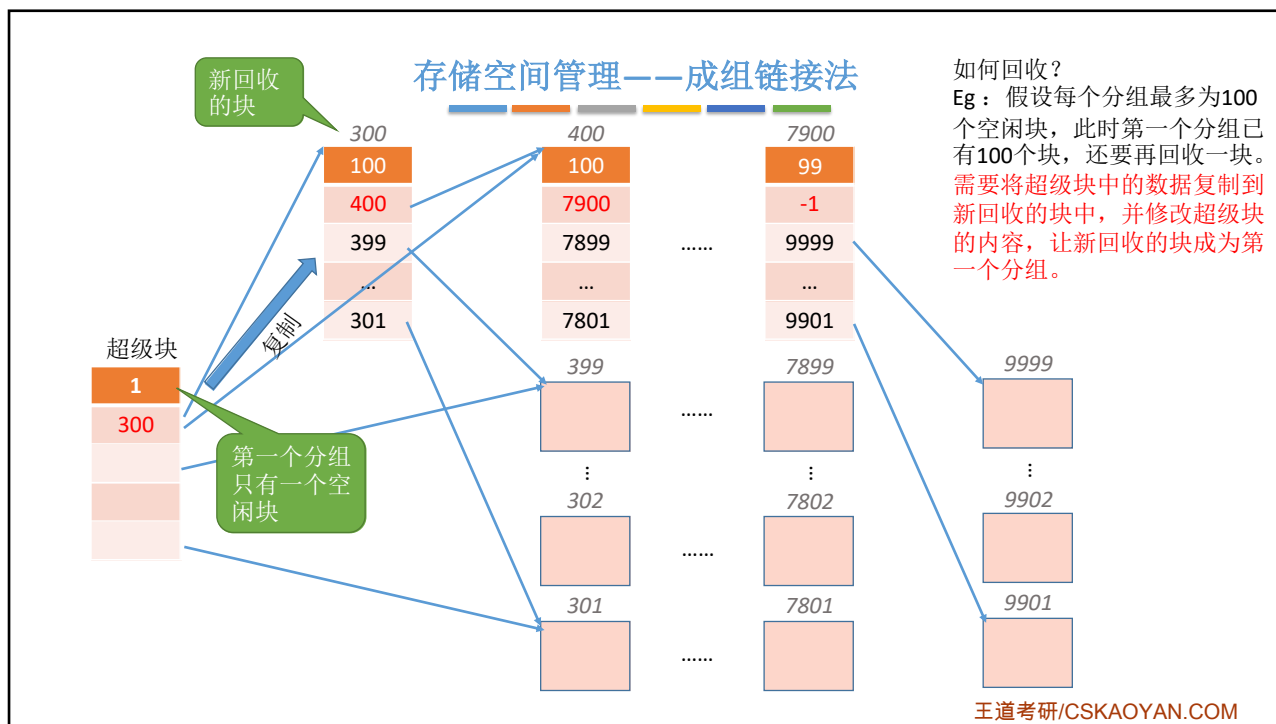
2020/5/20



15



16



17



18