物理设计(索引与散列)

单 位：重庆大学计算机学院

- excel文档记录数量很多的情况下，如何快速查找需要的记录？

915001016992708 47R	1	重庆市辉煌会展服务有限公司	正常	699270847	查账	何纲	居民身份证	511202197507212 27X	13709458616 13709458616
512201196607253722	1	万州区清堰坡姜妹美发室	正常		核定	姜发菊	居民身份证	512201196607253722	64885484 64885484
91500101577165 31XC	1	重庆市万州区莱洪建材经营部	正常	57716531X	核定	张华英	居民身份证	512221196208180023	
915001013049354629	1	江西洪明建筑工程有限公司重庆万州分公	正常	304935462	查账	刘斌龙	居民身份证	500235198601134879	18166375858
91500101304976934Y	1	重庆市乾元房地产经纪有限公司	正常	304976934	查账	杨海琼	居民身份证	510214198203221221	18996557788
410728687117247	1	河南宏新防腐安装有限公司	报验		查账	赵峰	居民身份证	410728197910123097	
500101584289556	1	重庆市万州区焕华装饰材料经营部	正常	584289556	核定	史焕华	居民身份证	512201197210190318	
512224196906127819	1	向阳华	正常		核定	向阳华	居民身份证	512224196906127819	64868256
915001013204017270	1	重庆市万州区铭雨明珠酒店有限公司	正常	320401727	查账	雷育鸣	居民身份证	512225196904199540	
500101207947234	1	重庆市万州区江南建筑工程有限公司（天	报验	207947234	查账	陈来虎			13896355226 13896355226
511202197502211315 01	1	万州区沙龙路美卡旧车信息服务部	正常		核定	王学军	居民身份证	511202197502211315	
512201196803054325	1	万州区周家坝艺瑞印刷厂	正常		核定	李天琼	居民身份证	512201196803054325	
50010345041610X	1	重庆市金建科技发展有限公司	报验	45041610X	查账	杨小英	居民身份证	510202195302161229	63633883 63633883
91500101569920020E	1	重庆慧忠建筑劳务有限公司	正常	569920020	查账	张明忠	居民身份证	512221197101101932	18996689999
915001017815615125	1	重庆市天鸿物流集团有限公司汽车交易市	正常	781561512	查账	万本斌	居民身份证	512221197012121392	58359166 13709438178
500101586870045	1	重庆市万州区继容美容美发店	正常	586870045	核定	龚继容	居民身份证	512201196710193748	15923457225 15923457225
91500101588023536T	1	重庆市万州区曾永刚水产养殖场	正常	588023536	核定	曾永刚	居民身份证	512221197502062495	
511202196707142531	1	重庆市万州区鑫琦文印部	正常		核定	陶于生	居民身份证	511202196707142531	64885938 64885938
512221550922511	1	钟广才	正常		核定	钟广才	居民身份证	512221550922511	58744722
915001012876203374	1	重庆市万州区渝东机电设备有限公司	正常	287620337	查账	刘艳蓉	居民身份证	512201197110021920	58116439 13908268227
50010119871026979X	1	刘太平	正常		核定	刘太平	居民身份证	50010119871026979X	
500101198602163 71X	1	万州区周家坝鸿强汽车维修部	正常		核定	莫家祥	居民身份证	500101198602163 71X	
91500101742896117Y	1	重庆市万州区宏润商贸有限公司	正常	742896117	查账	杨林	居民身份证	511202197411141614	58377560 13983513097
915001013049811 27D	1	重庆市愉旅交通设施有限公司	正常	304981127	查账	易海涛	居民身份证	512221196809030055	13330311379
512221197202072 78X01	1	万州区清明路郎姐副食经营部	正常		核定	郎华蓉	居民身份证	512221197202072 78X	
915001015801712350	1	重庆市万州区唐军种子经营部	正常	580171235	核定	唐军	居民身份证	511203197510120276	13996555888
512221196907253 99X01	1	刘乾文	正常		核定	刘乾文	居民身份证	512221196907253 99X	85798417
500101060515276	1	重庆市万州区张安明花椒种植园	正常	060515276	核定	张安明	居民身份证	512221195207161299	
500101573416820	1	重庆市万州区孙渝童装经营部	正常	573416820	核定	孙勇	居民身份证	511224197404241017	58808816 15870599699
512201197212303 3401	1	万州区春天花园洪钢家电经营部	正常		核定	程洪钢	居民身份证	512201197211230334	15084463966
91500101595168549A	1	重庆市缘达医药有限公司	正常	595168549	查账	邱军闵	居民身份证	512222197212068051	58108021 13509470760 1
500101573401434	1	重庆市金鼎煤业有限公司干坝子煤矿	正常	768860358	查账	高艮贤	居民身份证	512221621023161	13452607888
512223197312133275	1	石红权	正常		核定	石红权	居民身份证	512223197312133275	13368481168 13368481168
92500101MA5UHXXH58	1	黎长燕	正常		核定	黎长燕	居民身份证	512221197312053504	
915001015889058 11G	1	重庆市万州区付绍英畜禽销售部	正常	588905811	核定	付绍英	居民身份证	511202197011153886	
512221195701217278	1	何天良	正常		核定	何天良	居民身份证	512221570121727	无 无
512221196510125113	1	吴鉴荣	正常		核定	吴鉴荣	居民身份证	512221196510125113	13983122434 13983122434
512201197207262544	1	张容	正常		核定	张容	居民身份证	512201197207262544	85784525 85784525
500101195802160 91501	1	瑞清汽配	正常		核定	张瑞清	居民身份证	500101195802160915	58960409 13996626429
500101586895023	1	重庆市万州区彩装服饰经营部	正常	586895023	核定	陈玲	居民身份证	500101198608093 02X	87508509 13638298284
500903450412491	1	重庆渝先通信设备有限公司	报验	450412491	查账	吴勇	居民身份证	510213690403205	68692882 13436008166

主要学习目标

- 索引的种类
- 索引的基本概念和应用

思考问题

- 数据库索引如何创建，SQL语句是什么？

一 索引简介

讨论1. 数据库索引及其作用？

1)到底什么是(数据库)索引?

(数据库)索引是一种与(数据库)文件相关联的附加结构，额外增加的一个辅助文件！ P. 268

数据库系统中的索引与图书馆中书的索引所起的作用一样。例如，为了根据给定 *ID* 检索一条 *student* 记录，数据库系统首先会查找索引，找到相应记录所在的磁盘块，然后取出该磁盘块，得到所需的 *student* 记录。

显然：在满足条件的记录数较少(这种应用占多数情况)时，索引的效果才明显！

在存储了数千条学生记录的大型数据库中，维护一个排序的学生 *ID* 列表的效果并不好，因为索引本身将非常大；而且，即使通过排序的索引减少了搜索时间，查找一个学生也仍然是非常费时的工作。我们可以使用更复杂的索引技术来作为替代。我们将在本章讨论几种这样的技术。

有两种基本的索引类型：

- **顺序索引**。基于值的顺序排序。
- **散列索引**。基于将值平均分布到若干散列桶中。一个值所属的散列桶是由一个函数决定的，该函数称为散列函数(hash function)。

2) 索引的作用和类型?

3) 什么是搜索码和索引项(索引记录)?

搜索码：用于在文件中查找记录的属性/属性组； p. 268

索引项：由一个搜索码值和指向具有该搜索码值的一条/多条记录的指针构成。 P. 269

(索引项/索引记录是构成索引结构/索引文件中的基本要素)

二 顺序索引

讨论2. 顺序索引及其特点？

1. 顺序索引的基本概念 p. 269

1)什么是顺序索引，有哪些不同类型?

图11-1 (instructor)记录的顺序文件

顺序索引：基于搜索码值的顺序排序（指索引项在索引文件中）

主索引：索引文件排序与数据文件排序相同（只能有一个）

辅助索引：索引文件排序与数据文件排序不相同（可多个）

被索引文件中的记录自身也可以按照某种排序顺序存储，正如图书馆中的书按某些属性(如杜威十进制数)顺序存放一样。一个文件可以有多个索引，分别基于不同的搜索码。如果包含记录的文件按照某个搜索码指定的顺序排序，那么该搜索码对应的索引称为**聚集索引**(clustering index)。聚集索引也称为**主索引**(primary index)；主索引这个术语看起来是表示建立在主码上的索引，但实际上它可以建立在任何搜索码上。聚集索引的搜索码常常是主码，尽管并非必须如此。搜索码指定的顺序与文件中记录的物理顺序不同的索引称为**非聚集索引**(nonclustering index)或**辅助索引**(secondary index)。常用术语"聚集的"(clustered)和"非聚集的"(nonclustered)来代替"聚集"(clustering)和"非聚集"(nonclustering)。

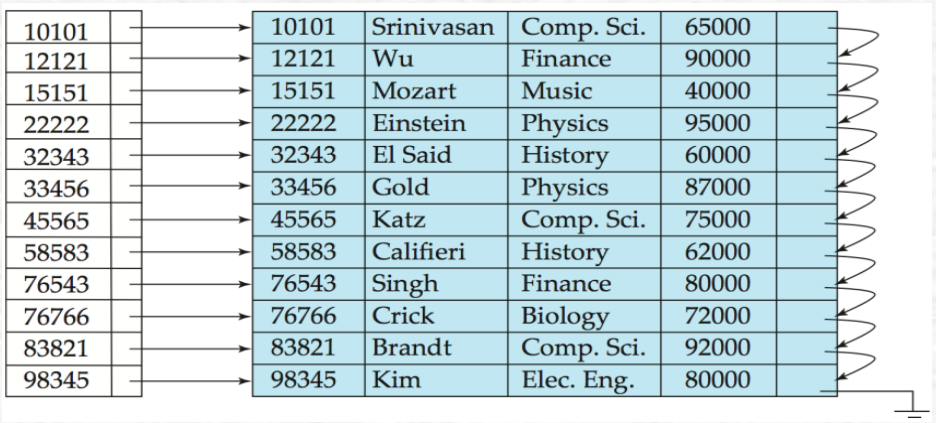
2. 稠密索引和稀疏索引

p. 269

2)稠密索引与稀疏索引有何不同?

稠密索引：索引文件中，每个搜索码都有一个索引项

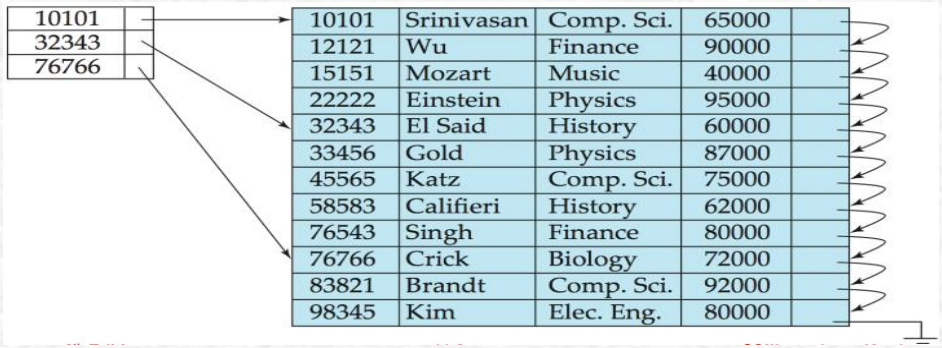
稀疏索引：索引文件中，只为某些搜索码建立索引项



3) 索引项的次序必需与记录的次序相同吗?

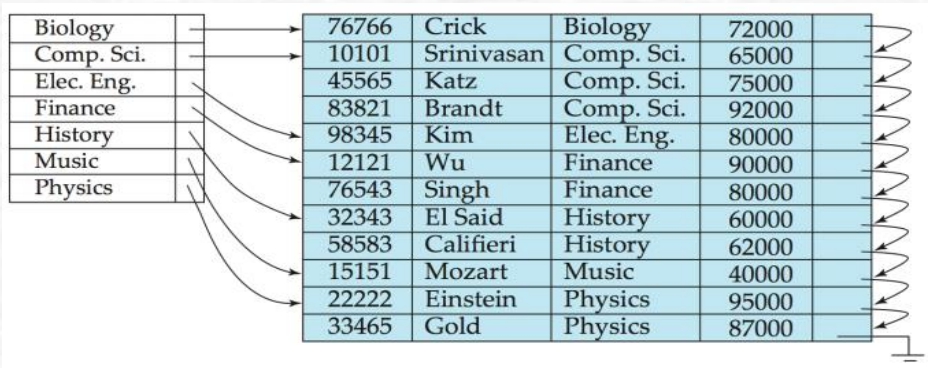
稠密索引:可次序不同.
图1中的次序相同，
因为它恰巧是主索引!

稀疏索引:次序需相同.
只有主索引才能使用!



4)稀疏索引相比稠密索引的好处 p.270?

1) 降低索引文件空间开销，
2) 提高搜索效率(跳跃查找)



5)(即使稠密)索引为何能加快查找效率?

1) 索引小，可在内存中处理
2) 索引排了序，查找效率高
3) 避免逐个读全部记录文件

3. 多级索引 p. 271

6)什么是多级索引，有何好处?

多级索引好处：进一步提高记录查找效率！

(1)稠密索引的空间占用分析：
1,000,000(百万)记录 R，索引占10,000块空间
(设1块(4KB)存放100条索引项) 40MB空间
100,000,000(1亿)元组,索引占1000,000块空间
4GB空间
若采用普通顺序索引,因主存无法容纳索引文件
一次搜索(搜索一个记录),需要多次读入磁盘块

(2)不同索引搜索效率分析：
即使采用二分法搜索：需读$\log_2(b)$个索引文件块
(设 b 为索引占用的块数)
b=10,000块(2的14次方),一次查询14次读块操作
(设一次读块操作需10毫秒，一次搜索140毫秒)
即1秒可执行约7次搜索

但用二级索引：因内层(下层)占10,000个磁盘块
故外层(上层)有10,000个索引项，占用100个块
(设外层的100块已放在主存中，因占空间小)
则一次查询只需读入1个索引块-而非14次读操作
(提高14倍)即1秒可执行100次搜索

多级索引：在索引文件上再建索引！

图 11-5 二级稀疏索引

(即使外层100块需临时读入，二分搜索需7次读操作)
则一次搜索共需1+7次读操作-也非14次读操作
即(一次搜索仅80毫秒)1秒可执行12次搜索

4. 辅助索引P. 273

7) 辅助索引有何特点?

辅助索引：必须是稠密索引！
对每个搜索码都有一索引项，
对每个记录有一个地址指针。

8) 辅助索引中，为何需要引入间接指针?

注:可建立多属性上的复合索引
（请自行举出一个实例）p. 273

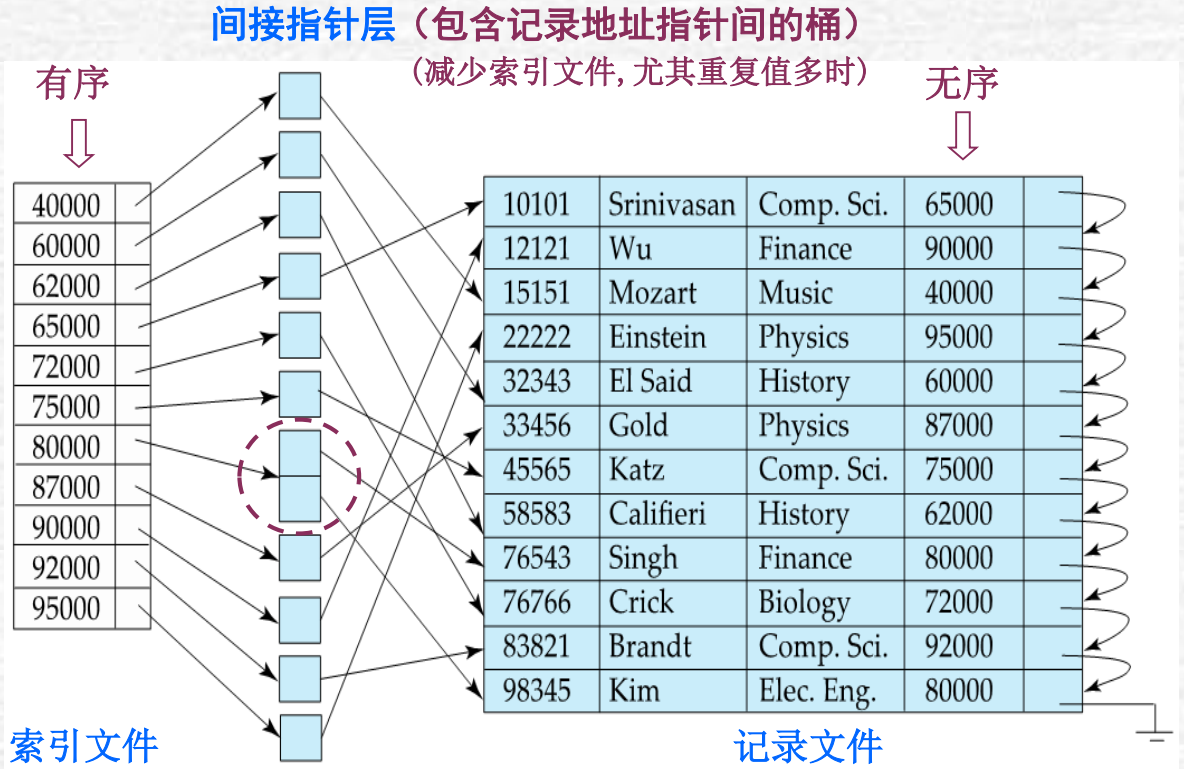


图11-6 instructor的辅助索引（基于非候选码salary-索引码）

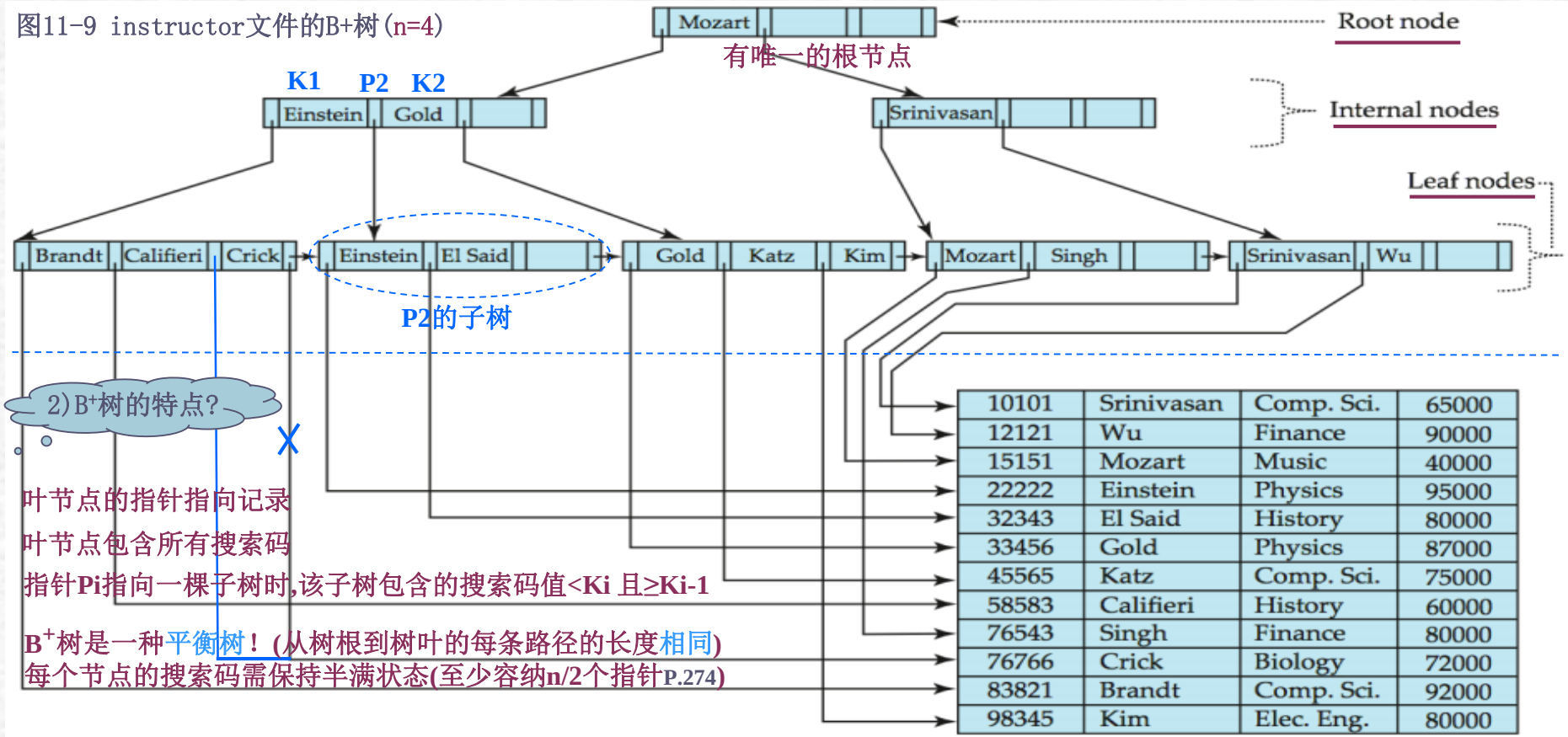
三 B+树索引*（讲解为主）

讨论3. B+树索引的结构及特点？

1. B+树索引的结构和特点 p. 274

1）B+树中包含哪些类型节点，节点的结构？

图11-9 instructor文件的B+树(n=4)



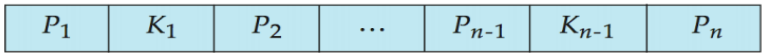


图11-7 B+树的节点(结构)

包含n个(应用需要设定)指针和n-1个搜索码，搜索码依序排列。
指针 Pi (内部节点)指向一棵子树，或者(叶节点)直接指向记录。

2. B+树的查询

p. 275

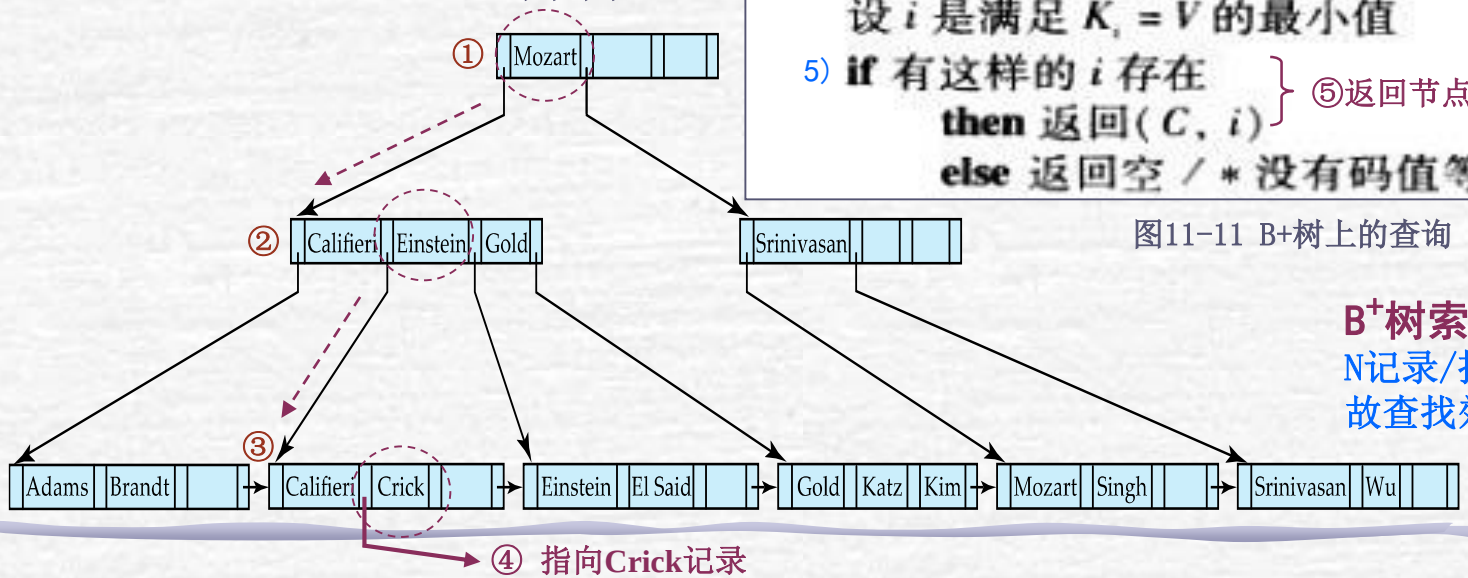
第1次循环,执行1)和4):
Ki=Mozart,i=1,C=P1→②节点

第2次循环,执行1)和4):
Ki=Einstein,i=2,C=P2→③节点

第3次循环,执行1)和3):
Ki=Crick,已是叶节点,i=2,C=P2→ ④记录指针

最后的if语句5):
Ki=Crick,i=2,返回P2--⑤指向Crick记录

V=Crick查询示例：



Function find(value V)　V=Crick

置 C = 根结点　C--①根节点
while C 不是叶结点 begin
1) 令 i = 满足 V≤C.K_i 的最小值
2) if 没有这样的 i then begin
　　令 P_m = 结点中最后一个非空指针
　　置 C = C.P_m
　end
3) ----▸ else if (V = C.K_i)
　　then 置 C = C.P_{i+1}
4) ----▸ else C = C.P_i /* V < C.K_i */
　end
　/* C 是叶结点 */
　设 i 是满足 K_i = V 的最小值
5) if 有这样的 i 存在
　　then 返回(C, i)　} ⑤返回节点③和指向Crick的指针
　　else 返回空 /* 没有码值等于 V 的记录存在 */

3)如何基于B+树索引快速查找记录?

图11-11 B+树上的查询

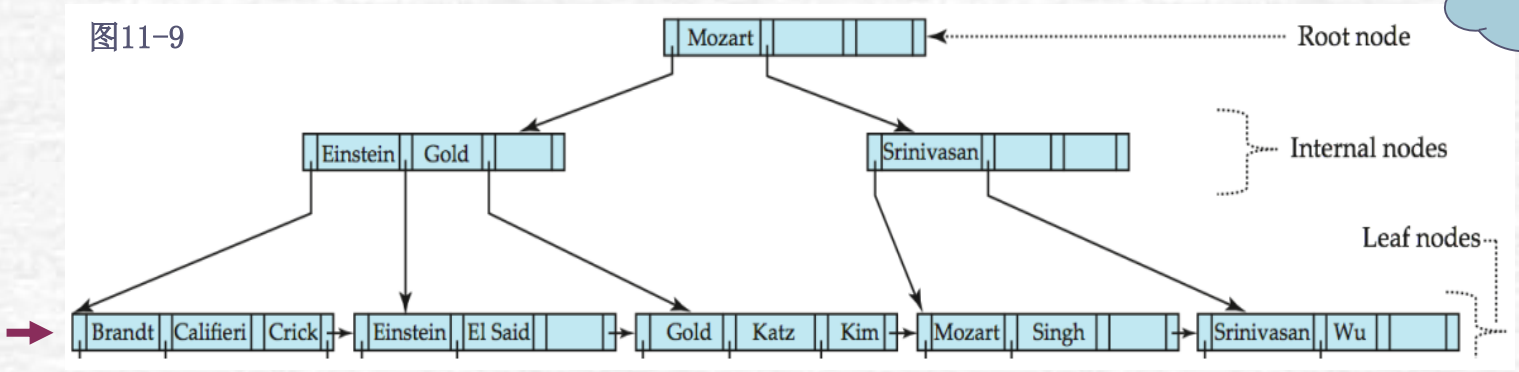
B+树索引特点:查询效率极高
N记录/搜索码总数,树高$\log_{\lceil n/2 \rceil} N$
故查找效率(读块次数):对数时间
p. 282

3. B+树的插入更新 p. 278

4)如何在B+树上插入一个搜索码?

在图11-9的B+树中插入搜索码Adams的例示:

图11-9



节点未满时，只需直接将搜索码填写到相应节点

原节点已满，需‘一分为二’(各放一半搜索码)

上层节点未存两新节点指针，需增加一个索引码
按照搜索码与子树上搜索码的关系,为Califieri
(右子树上搜索码中的最小者)

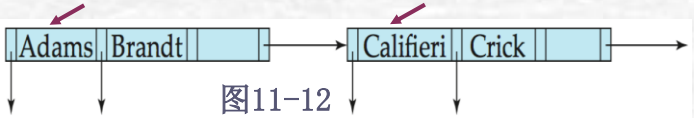
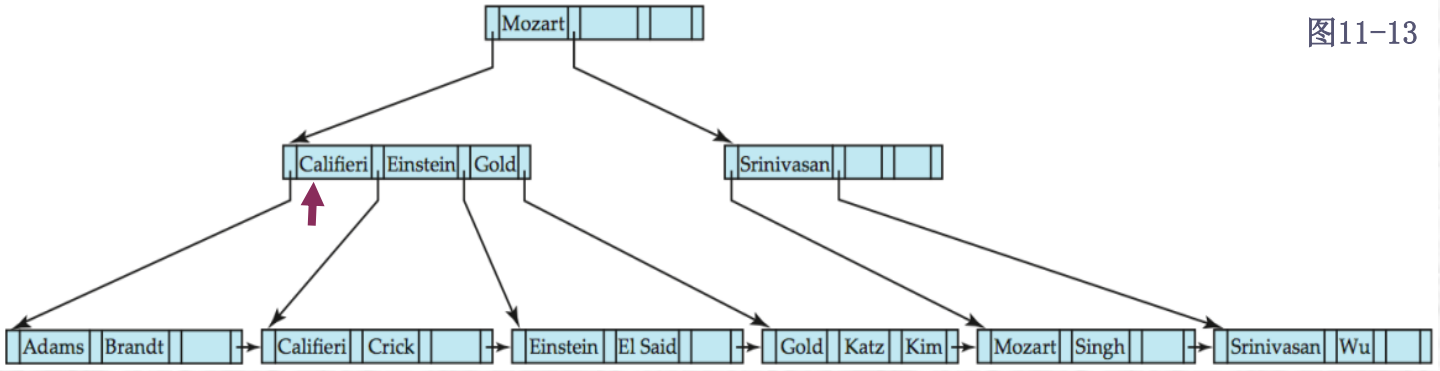


图11-12

B+树插入特点：开销小(仅极小部分有变化)

图11-13

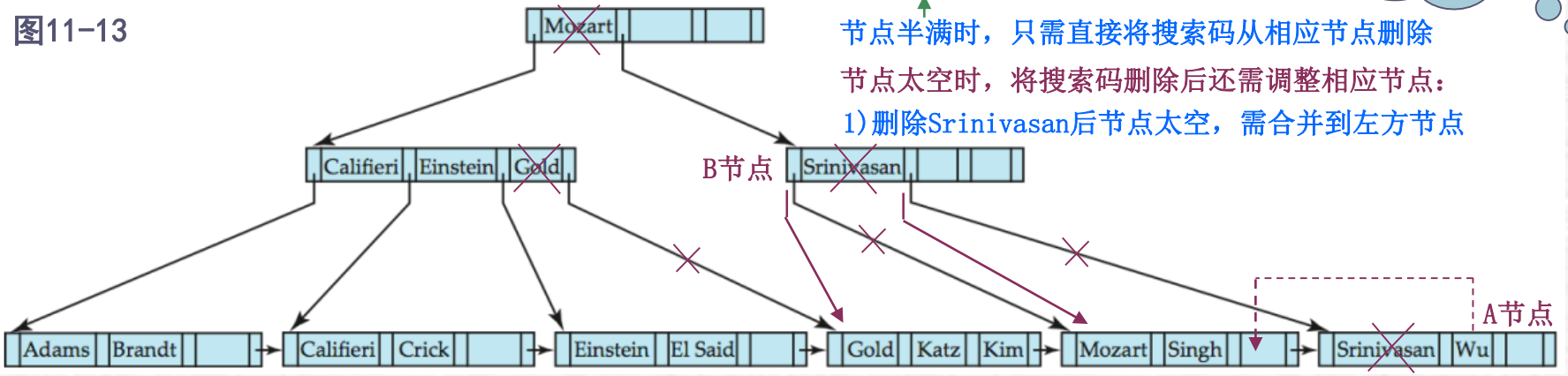


4. B+树的删除更新 p. 280

5)如何在B+树上删除一个搜索码?

在图11-13的B+树中删除搜索码Srinivasan的例示:

图11-13



指针数≧n/2 (n=4)

节点半满时，只需直接将搜索码从相应节点删除

节点太空时，将搜索码删除后还需调整相应节点：

1)删除Srinivasan后节点太空，需合并到左方节点

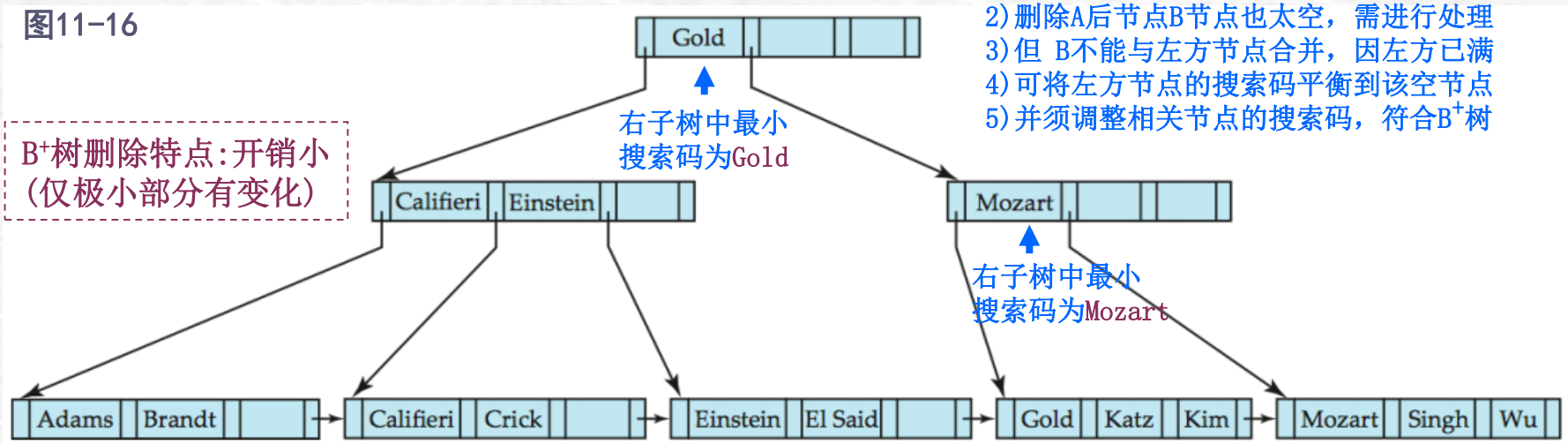
2)删除A后节点B节点也太空，需进行处理

3)但 B不能与左方节点合并，因左方已满

4)可将左方节点的搜索码平衡到该空节点

5)并须调整相关节点的搜索码，符合B+树

图11-16



B+树删除特点:开销小
(仅极小部分有变化)

四 **散列索引(讲解为主)

讨论4. 散列索引及其特点？

例中散列函数：IDmod8(对8取模)
设计散列函数：散列值分布均匀

桶数：N（例子中为8个桶）
(1个散列桶, 为一个磁盘块, p. 288
但也可以小于或大于一个磁盘块)

溢出桶(可能多个, 尤其不均匀时)
当某桶装满时，存储溢出索引项

插入一个索引项：
计算搜索码的散列值确定桶
然后在相应桶中写入索引项

删除一个索引项：
计算搜索码的散列值确定桶
然后在相应桶中删除索引项

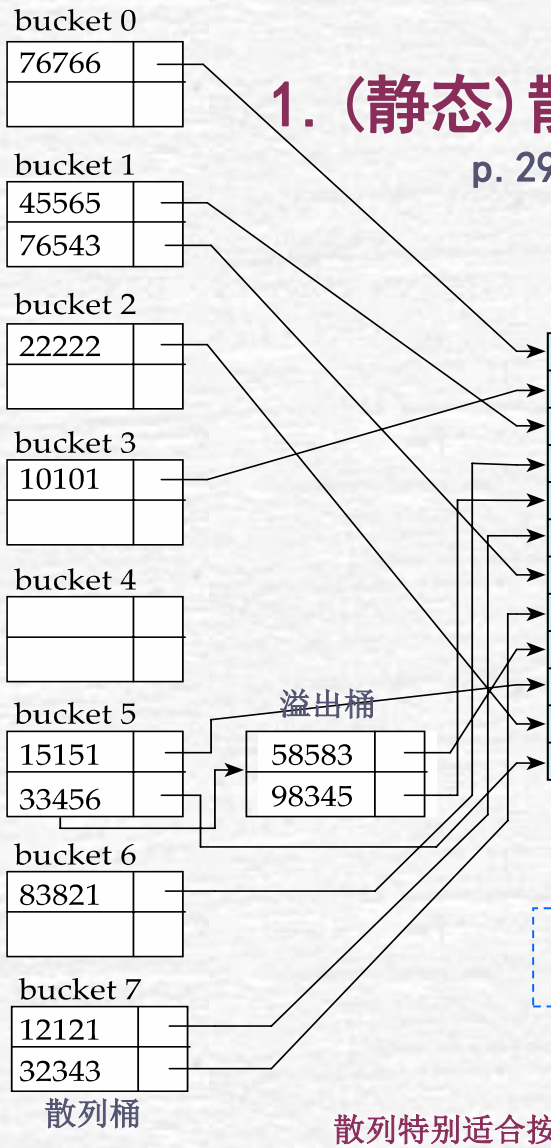
查找记录：
计算搜索码的散列值确定桶
然后在相应桶中得到索引项
根据索引项中指针得到记录

1. (静态)散列索引

p. 291

1)什么是散列索引，主要特点?

2)如何插入/删除一个索引项?



76766	Crick	Biology	72000
10101	Srinivasan	Comp. Sci.	65000
45565	Katz	Comp. Sci.	75000
83821	Brandt	Comp. Sci.	92000
98345	Kim	Elec. Eng.	80000
12121	Wu	Finance	90000
76543	Singh	Finance	80000
32343	El Said	History	60000
58583	Califieri	History	62000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000
33465	Gold	Physics	87000

↑搜索码值

图11-25Instructor文件在搜索码ID上的散列索引

散列索引：采用散列函数将搜索码映射到散列桶
通过散列索引，支持基于搜索码的记录快速查找

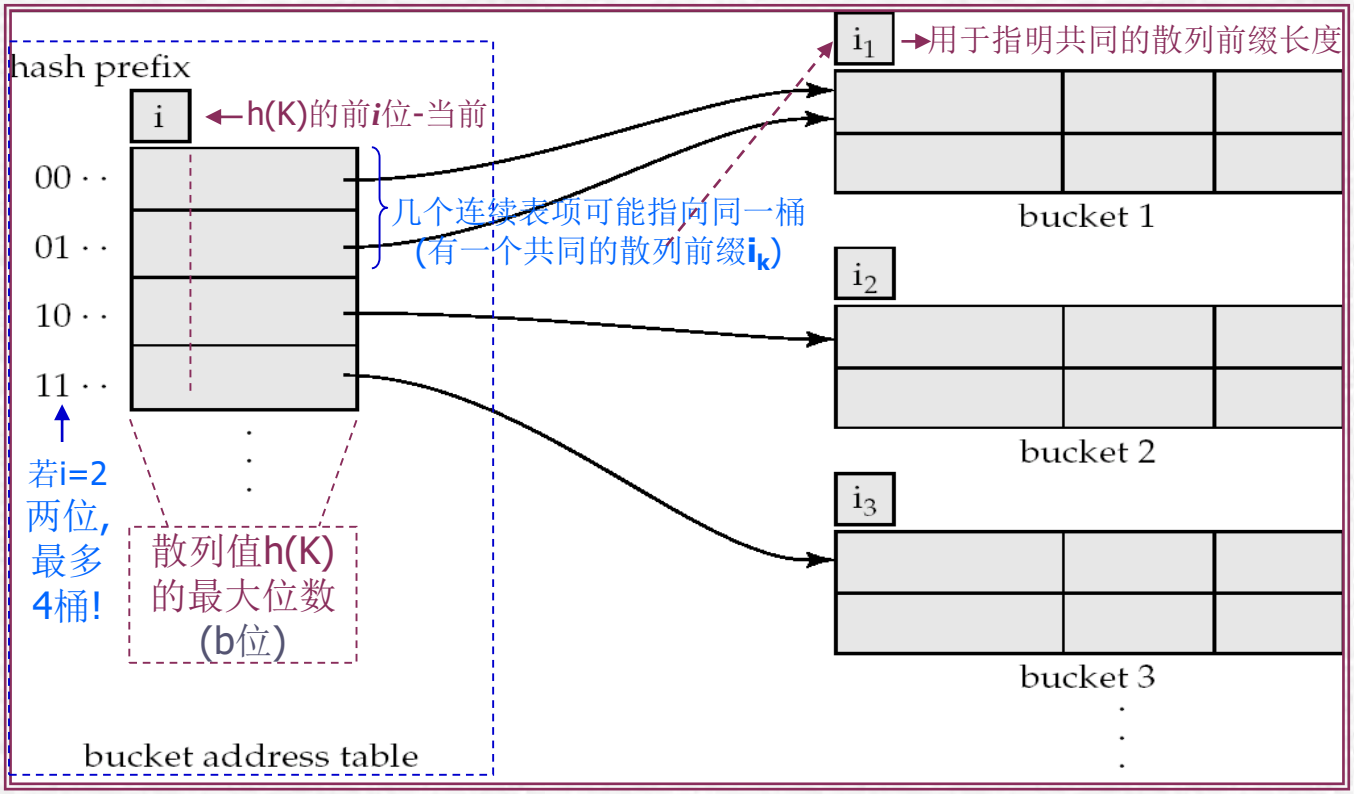
3)如何查找记录?

散列特别适合按搜索码的等值查找!

2. 动态散列 p. 292

4)什么是动态散列，主要特点?

散列桶数目不固定，通过散列前缀i控制，开始非常少，随索引项增加而逐渐增大!



一般地$i_k \le i$

In this structure, $i_2 = i_3 = i$ (=2), whereas $i_1 = i - 1$ (=1) 而 i=2

(**see next slide for details**)

5) 如何在动态散列中定位和插入记录？

Use of Extendable Hash Structure

-定位和插入记录

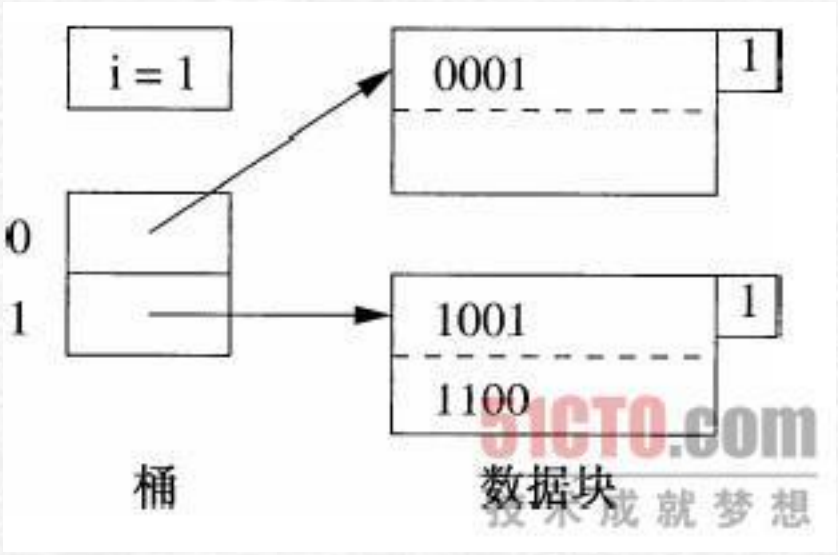
- **如何知道(桶地址表)指向同一桶:** Each bucket *j* stores a value i_j 正整数
  - All the entries that point to the same bucket have the same values on the **first i_j bits**.
- **如何定位属于哪一桶：** To locate the bucket containing search-key K_j:
  1. Compute $h(K_j) = X$ *即计算该搜索码K_j相应的二进数表示*
  2. Use the first *i high order bits* of *X* as a displacement偏转/位移 into bucket address table, and follow跟随 the pointer to appropriate bucket

  桶地址表中指向桶j的表项编号(的计算公式)为: $2^{(i-i_j)}$
- **如何插入记录：** To insert a record with search-key value K_j
  - follow same procedure as look-up and locate the bucket, say *j*. (桶j)
  - **If** there is room in the bucket *j*, insert record in the bucket.
  - **Else** the bucket must be split and insertion re-attempted 重试 (will see in next slide)
    - Overflow buckets used instead in some cases (will see in next slide)

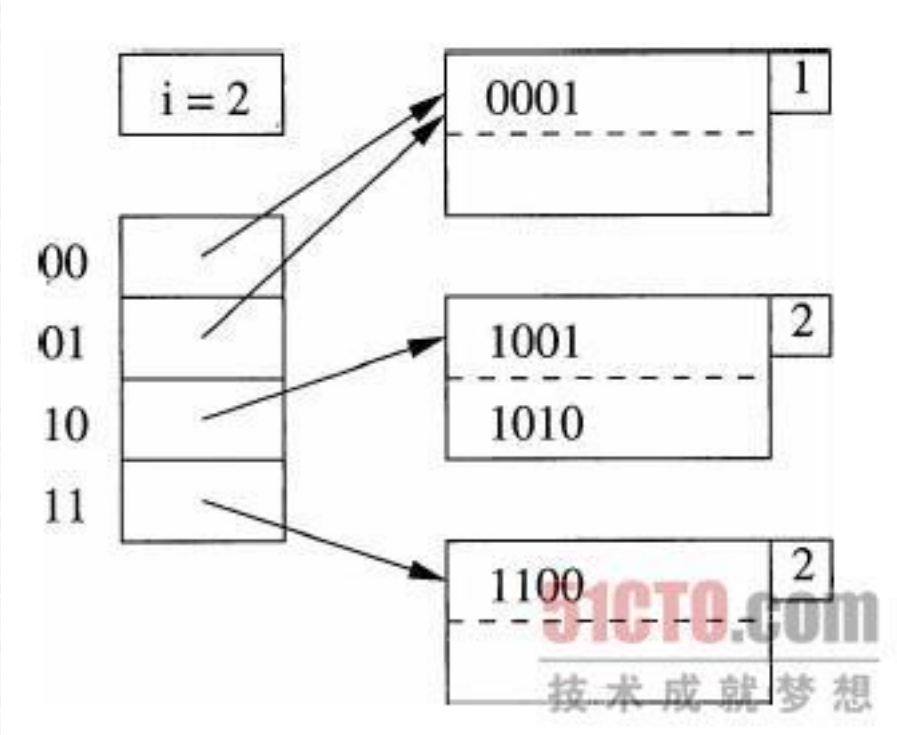
散列函数h为每个键计算出一个K位二进制序列，该K足够大，比如32。但是，桶的数目总是使用从序列第一位或最后一位算起的若干位，此位数小于K，比如说是i位。也就是说，当i是使用的位数时，桶数组将有2^i个项。

假定K=4，即散列函数h只产生4位二进制序列。当前使用的只有其中一位，正如桶数组上方的框中i=1所标明的那样。因此，桶数组只有两个项，一个对应0，一个对应1。

假如我们插入一个键值散列为1010序列的记录。



我们在前面图表中插入一个键值散列为**1010**序列的记录。因为第一位是**1**，所以该记录属于第二个块。然而，该块已满，因此需要分裂。这时我们发现**j=i=1**，因此我们首先需要将桶数组加倍，如图所示。图中我们已将**i**设为**2**。



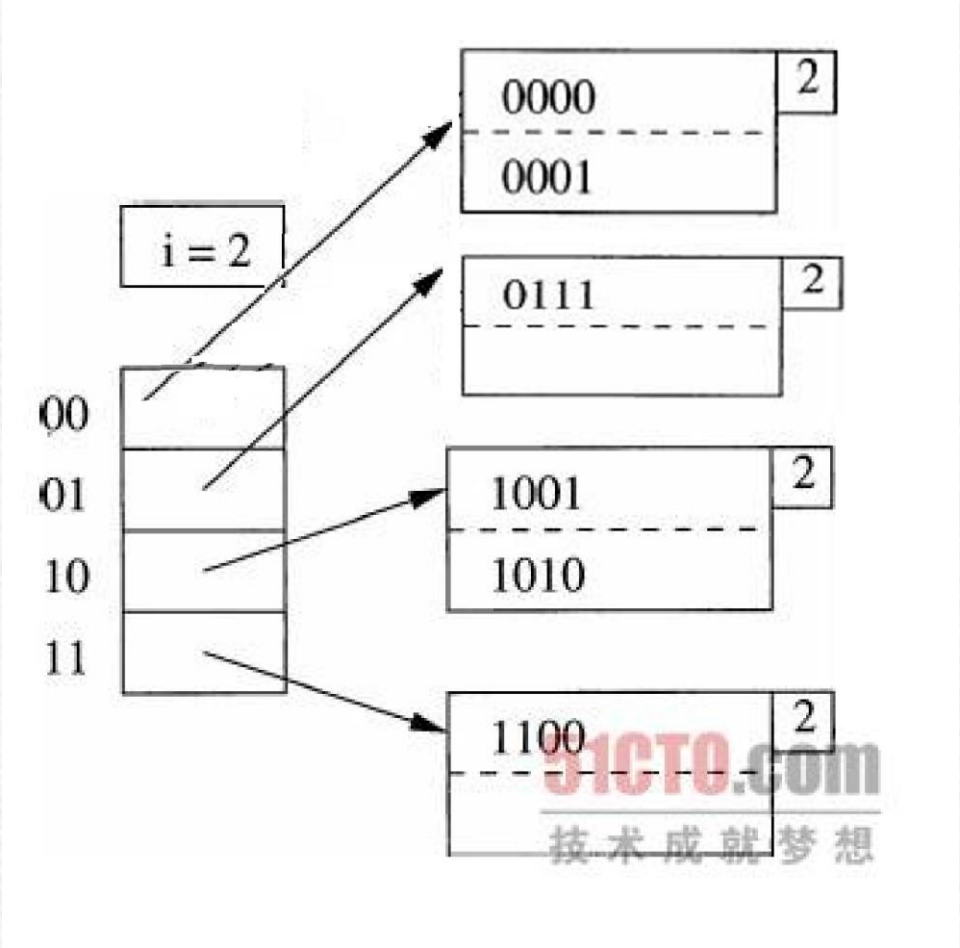
现在，假定我们来插入键值分别列为**0000**和**0111**的记录。

插入键值分别列为0000和0111的记录

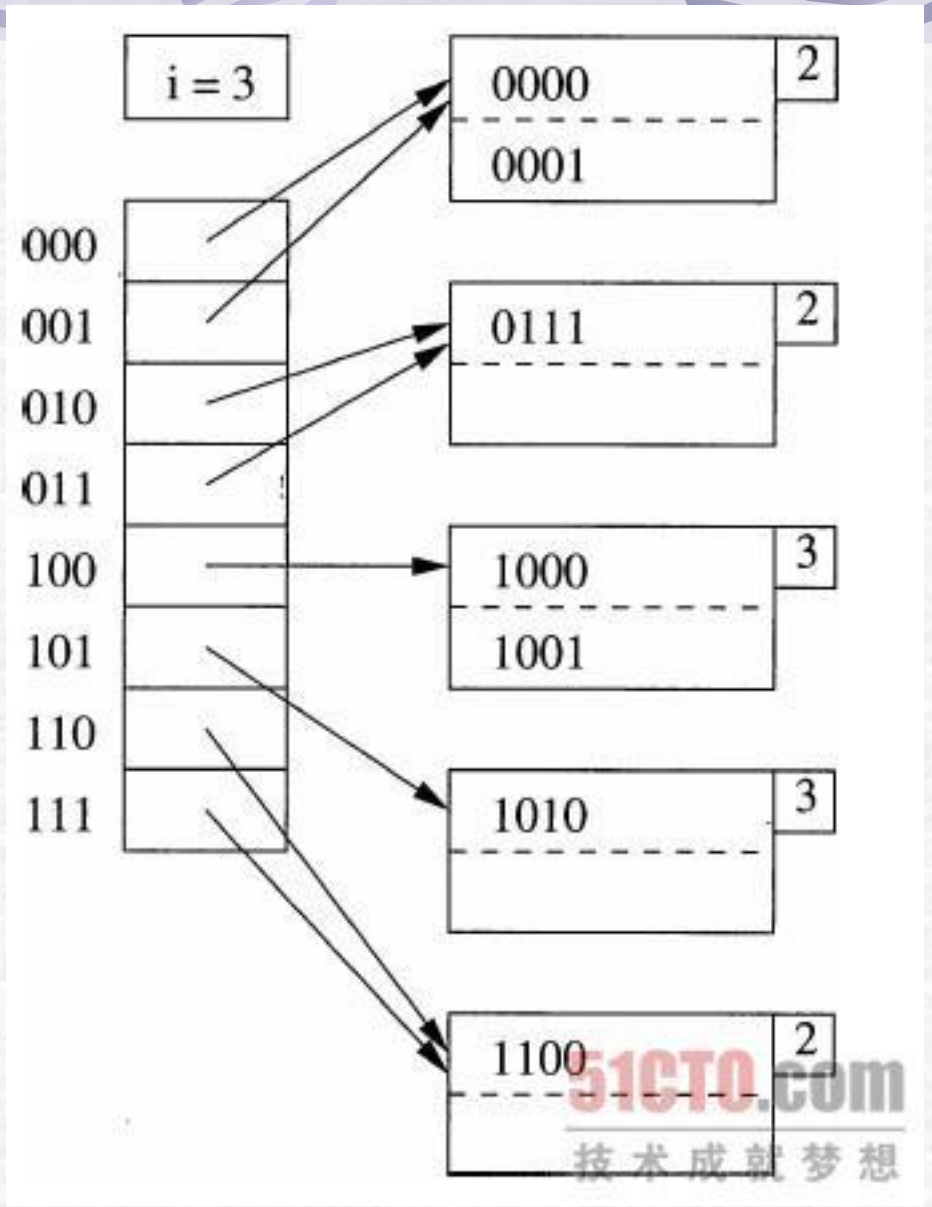
我们插入键值分别列为0000和0111的记录。这两个记录都属于图中第一个存储块，于是该块溢出。因为该块中只用一位来确定其成员资格。

而i=2，所以我们就不用调整桶数组。我们只需分裂该块，让0000和0001留在该块，而将0111存放到新块中，桶数组中01项改为指向新块

插入一个键值为1000的记录



插入一个键值为**1000**的记录



随堂小测试

- 简述稀疏索引和稠密索引的优缺点及应用场景？
- 散列索引和顺序索引的区别是什么？

课堂总结和作业安排

- 基本知识：
  - 顺序索引
  - B+树索引
  - 散列索引
- 延展性学习：
  - 是否有更好的索引技术？
- 作业

  第11章习题：11.3,11,6，11.17 a)