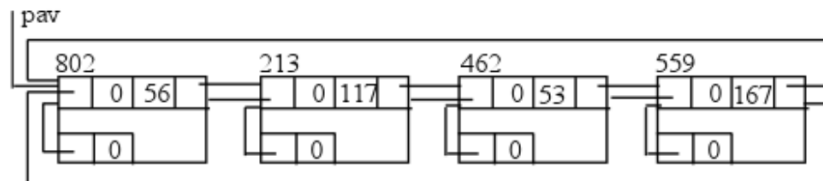


第八、九、十章作业

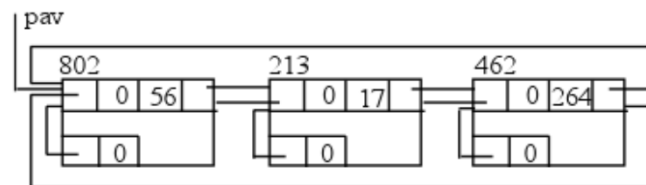
钟赞 2016K8009915009

8.1 解：

(1) 系统回收一个起始地址为 559，大小为 45 的空闲块之后，与右侧起始地址 604 的空闲块合并为起始地址 559，大小为 167 的空闲块：

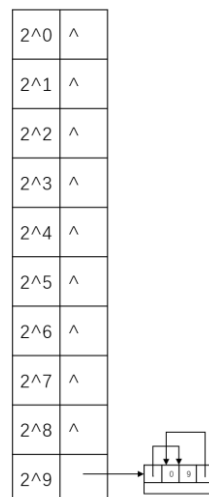


(2) 系统首先将起始地址为 213 大小为 117 的空闲块分配给 100 存储块请求；然后将起始地址为 515 大小为 44 的空闲块与它左右的空闲块合并为起始地址为 462，大小为 264 的块：



8.7 解：

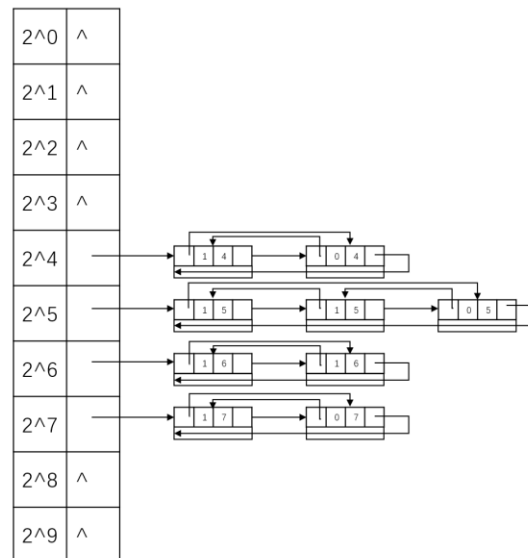
(1) 初始情况：



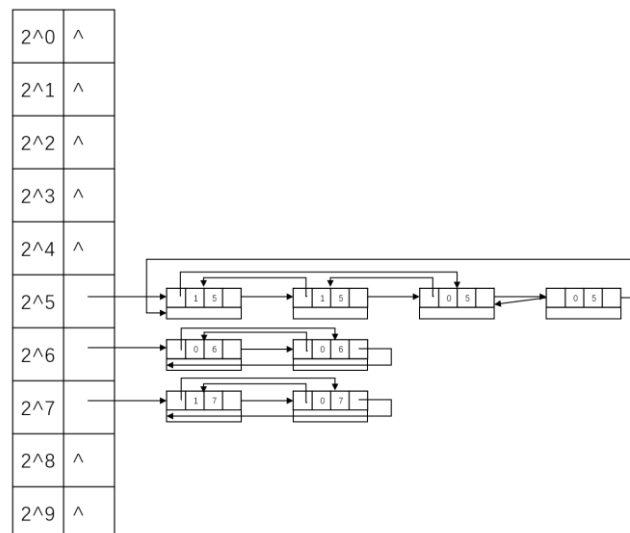
(2) 每个用户所得存储快情况：

用戶存儲大小	起始地址
23	0
45	64
52	128
100	256
11	32
19	192

六个用户进入后链表状态：



(3) 回收后链表状态

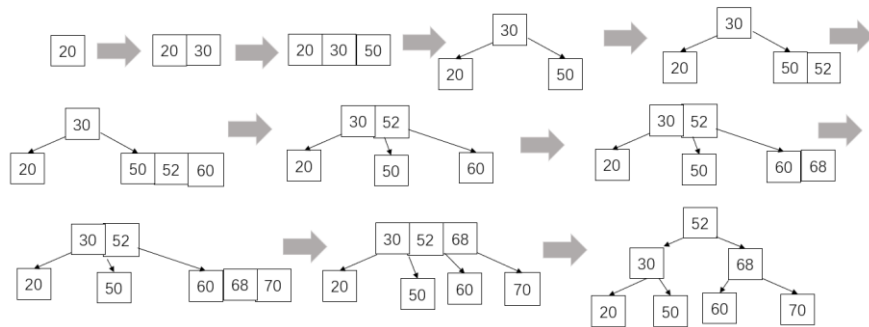


9.1 解：

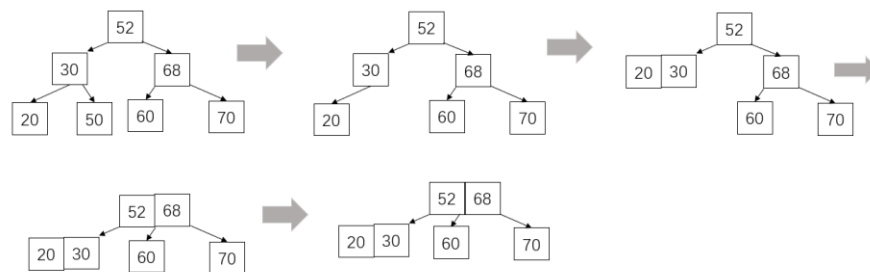
- (1) 相同。有序顺序表和无序顺序表： $n+1$
- (2) 相同。有序顺序表和无序顺序表： $(n+1)/2$
- (3) 不同。对于有序顺序表，找到第一个与 K 值相同的关键字后，只需再找到与 K 不同的元素即可停止查找；对于无序顺序表，需要一直查找到最后一个关键字。

9.14 解：

创建过程：



删除过程：



9.19 解：

$H(22) = 0$;

$H(41) = 2$;

$H(53) = 5$;

$H(46) = 6$;

$H(30) = 2$, 冲突 ; $H_1(30) = 2 + 1 \times 1 = 3$;

$H(13) = 6$, 冲突 ; $H_1(13) = 6 + 1 \times 2 = 8$;

$H(01) = 3$, 冲突 ; $H_1(01) = 3 + 1 \times 8 = 11$, 冲突 ; $H_2(01) = 11 + 1 \times 16 = 27$, 冲突 ; $H_3(01) = 27 + 1 \times 64 = 91$, 冲突 ; $H_4(01) = 91 + 1 \times 256 = 347$, 冲突 ; $H_5(01) = 347 + 1 \times 1024 = 1371$;

$H(67) = 3$, 冲突 ; $H_1(67) = 3 + 1 \times 8 = 11$, 冲突 ; $H_2(67) = 11 + 1 \times 16 = 27$;

哈希表如下：

哈希地址	0	1	2	3	4	5	6	7	8	9	10
关键字	22	67	41	30		53	46		13		01
次数	1	3	1	2		1	1		2		6

$$ASL = (1 \times 4 + 2 \times 2 + 3 + 6) / 8 = 17/8$$

9.24 解：

(1) 在校人数极端情况：本科生 = $4 \times 25 \times 200$ ，研究生 = $4 \times 25 \times 50$ ，总共 25000 人，装载因子 $\alpha=1$

(2) 学号 $C = C_1 C_2 C_3 C_4 C_5 C_6 C_7 C_8$

$$H1(C) = \begin{cases} C_{678} + C_{34} \times 200 + (C_{12} - 96) \times 200 \times 25 & (IF C_5 = 0) \\ 20000 + C_{678} + C_{34} \times 50 + (C_{12} - 96) \times 50 \times 25 & (IF C_5 = 1) \end{cases}$$

可以看出聚簇性很好。

(3) $\alpha < 1$ ，但是无法找到不冲突的 $H1$

10.1 解：

503, 087, 512, 061, 908, 170, 897, 275, 653, 426

直接插入排序：

第一趟：087, 503, 512, 061, 908, 170, 897, 275, 653, 426

第二趟：087, 503, 512, 061, 908, 170, 897, 275, 653, 426

第三趟：061, 087, 503, 512, 908, 170, 897, 275, 653, 426

第四趟：061, 087, 503, 512, 908, 170, 897, 275, 653, 426

第五趟：061, 087, 170, 503, 512, 908, 897, 275, 653, 426

第六趟：061, 087, 170, 275, 503, 512, 897, 908, 653, 426

第八趟：061, 087, 170, 275, 503, 512, 653, 897, 908, 426

第九趟：061, 087, 170, 275, 426, 503, 512, 653, 897, 908

希尔排序

($d[1]=5, d[2]=3, d[3]=1$)

第一趟：170, 087, 275, 061, 426, 503, 897, 512, 653, 908

第二趟：061, 087, 275, 170, 426, 503, 897, 512, 653, 908

第三趟：061, 087, 170, 275, 426, 503, 512, 653, 897, 908

快速排序

第一趟：(426, 087, 275, 061, 170) 503(897, 908, 653, 512)

第二趟：(170, 087, 275, 061) 426, 503(512, 653) 897(908)

第三趟：(061, 087) 170(275) 426, 503, 512(653) 897, 908

第四趟：061, 087, 170, 275, 426, 503, 512, 653, 897, 908

堆排序

建堆：061, 087, 170, 275, 426, 512, 897, 503, 653, 908

(输出 061) 087, 275, 170, 503, 426, 512, 897, 653

(输出 087) 170, 275, 512, 503, 426, 653, 897, 908

(输出 170) 275, 406, 512, 503, 908, 653, 897

(输出 275) 406, 503, 512, 897, 908, 653

(输出 406) 503, 653, 512, 897, 908

(输出 503) 512, 653, 908, 897

(输出 512) 653, 897, 908

(输出 653) 897, 908

(输出 897) 908

归并排序

第一趟：(087, 503)(061, 512)(170, 908)(275, 897)(426, 653)

第二趟：(061, 087, 503, 512)(170, 275, 897, 908)(426, 653)

第三趟：(061, 087, 170, 275, 503, 512, 897, 908)(426, 653)

第四趟：061, 087, 170, 275, 426, 503, 512, 653, 897, 908

基数排序

第一趟：170, 061, 512, 503, 653, 275, 426, 087, 897, 908

第二趟：503, 908, 512, 426, 653, 061, 170, 275, 087, 897

第三趟：061, 087, 170, 275, 426, 503, 512, 653, 897, 908

10.3 解：

- (1) 直接插入排序 稳定
- (2) 希尔排序 不稳定
- (3) 快速排序 不稳定
- (4) 堆排序 不稳定
- (5) 归并排序 稳定
- (6) 基数排序 稳定

10.15 解：

先通过完全二叉树类似进行两两比较，淘汰小的数，直到找出最大的数，需要比较 $n-1$ 次；

最小的数是被首次淘汰的数，至多有 $\lceil \frac{n}{2} \rceil + 1$ 个，从中找出最小的数至多需要 $\lceil \frac{n}{2} \rceil$ 次；

所以共需比较 $\lceil \frac{n}{2} \rceil + n - 1$ 次。

10.21 解：

以数列 4, 3, 2, 1 为例：

折半插入排序

第一趟：4, 3, 2, 1

第二趟：3, 4, 2, 1

比较 3, 4；

第三趟：2, 3, 4, 1

比较 2, 3；

第四趟：1, 2, 3, 4

比较 1, 3；比较 1, 2；

2-路归并排序法

第一趟：(3, 4) (1, 2)

比较 3, 4 ; 比较 1, 2

第二趟：(1, 2, 3, 4)

比较 3, 1 ; 比较 3, 2