9.3 哈希表

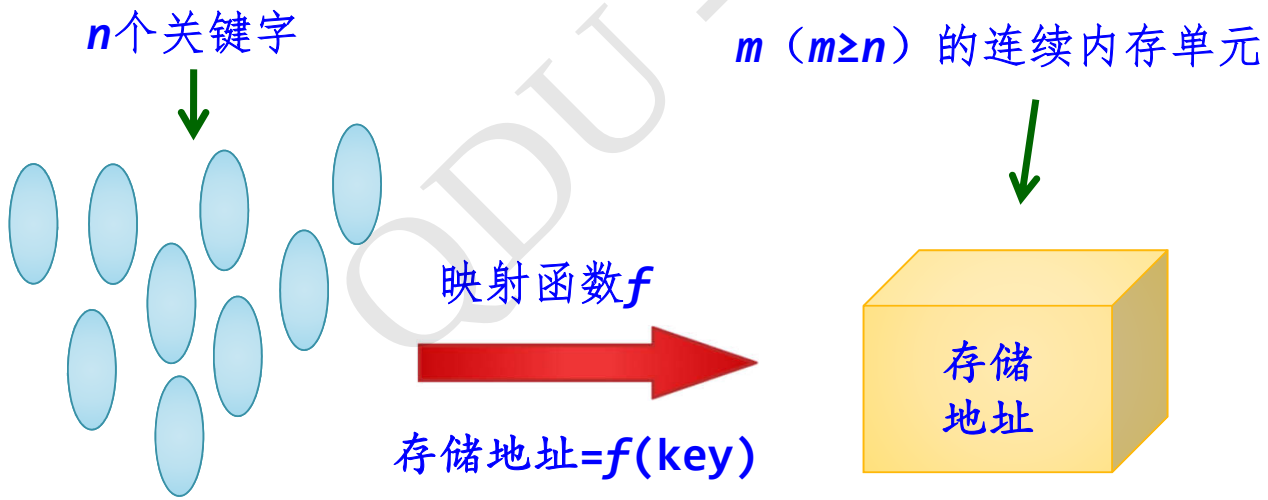
9.3.1 哈希表的基本概念

- 在前面讨论的各种结构(线性表、树等）中，记录在结构中的相对位置是随机的，和记录的关键字之间不存在确定的关系，因此，在结构中查找记录时需进行一系列和关键字的比较。这一类查找方法建立在“比较”的基础上。查找的效率依赖于查找过程中所进行的比较次数。

- 理想的情况是希望不经过任何比较，一次存取便能得到所查记录，为此，需在记录的存储位置和它的关键字之间建立一个确定的对应关系$f(key)$，由此，不需进行比较便可直接取得所查记录。在此，我们称这个对应关系$f(key)$为哈希（Hash）函数，或者散列函数，按这个思想建立的表为哈希表（散列表）。

9.3.1 哈希表的基本概念

- 哈希表（Hash Table）又称散列表，是除顺序存储结构、链式存储结构和索引表存储结构之外的又一种存储结构。



学生
成绩表

学号	姓名	分数
201201	王实	85
201205	李斌	82
201206	刘英	92
201202	张山	78
201204	陈功	90

地址	学号	姓名	分数
0			
1			
2			
3			
4			
5			

学生
成绩表

学号	姓名	分数
201201	王实	85
201205	李斌	82
201206	刘英	92
201202	张山	78
201204	陈功	90

- 哈希表长度m=6（存储单元的地址为0～5）
- 记录个数n=5
- 哈希函数：h(学号) = 学号-201201

地址	学号	姓名	分数
0	201201	王实	85
1	201202	张山	78
2			
3	201204	陈功	90
4	201205	李斌	82
5	201206	刘英	92

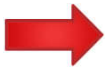
地址	学号	姓名	分数
0	201201	王实	85
1	201202	张山	78
2			
3	201204	陈功	90
4	201205	李斌	82
5	201206	刘英	92

哈希表
（哈希空间）

哈希函数：*h*(学号) = 学号-201201

查找201204的学生姓名

- *h*(201204)=201204-201201=3
- 地址3的记录姓名为"陈功"



O(1)，按关键字查找速度快

问题1

哈希函数：h(学号) = 学号-201201

- 如果n=30，m=35，但学号分布分散。
- 学号201201，h(201201)=201201-201201=0
- …
- 学号201250，h(201250)=201250-201201=49 ？

哈希函数：h(学号) = (学号-201201) % m

- 学号201201，h(201201)=(201201-201201)%35=0
- …
- 学号201250，h(201250)=(201250-201201)=49%35=14

问题2

哈希函数：*h*(学号) = (学号-201201) % *m*

- *h*(201204)=(201204-201201)%35=3
- 如果存在一个学号为201239

  h(201239)=(201239-201201)%35=38%35=3 ？

- 可能存在这样的问题，对于两个关键字k_i和k_j（$i \neq j$），有$k_i \neq k_j$（$i \neq j$），但$h(k_i)=h(k_j)$。把这种现象叫做发生了冲突，称k_i、k_j为同义词。
- 如果出现了冲突，后存储的记录会覆盖前面存储的记录。这是不允许的！

需要解决哈希冲突

哈希函数通常是一种压缩映象，所以冲突不可避免，只能尽量减少；当冲突发生时，应该有处理冲突的方法。

设计一个散列表应包括：

- ① 散列表的空间范围，即确定散列函数的值域；
- ② 构造合适的散列函数，使得对于所有可能的元素(记录的关键字)，函数值均在散列表的地址空间范围内，且出现冲突的可能尽量小；
- ③ 处理冲突的方法。即当冲突出现时如何解决。

9.3.2 哈希函数构造方法

构造哈希函数的目标：

- （1）使得到的哈希地址尽可能均匀地分布在*m*个连续内存单元地址上，所谓“均匀”(uniform)是指发生冲突的可能性尽可能最少。
- （2）同时使计算过程尽可能简单以达到尽可能高的时间效率。

- 根据关键字的结构和分布的不同，有多种构造哈希函数的方法。

1. 直接定址法

- 以关键字*key*本身或关键字加上某个数值常量c作为哈希地址的方法。即*h*(*key*)=*key*+c。

✓ 这种哈希函数计算简单，并且不可能有冲突发生。

- 当关键字的分布基本连续时，可用直接定址法的哈希函数；否则，若关键字分布不连续将造成内存单元的大量浪费。

学生
成绩表

学号	姓名	分数
201201	王实	85
201205	李斌	82
201206	刘英	92
201202	张山	78
201204	陈功	90

- 哈希表长度m=6（存储单元的地址为0～5）
- 记录个数n=5
- 哈希函数：h(学号) = 学号-201201

地址	学号	姓名	分数
0	201201	王实	85
1	201202	张山	78
2			
3	201204	陈功	90
4	201205	李斌	82
5	201206	刘英	92

2. 除留余数法

- 用关键字*key*除以某个不大于哈希表长度*m*的数*p*所得的余数作为哈希地址的方法。
- 除留余数法的哈希函数*h*(*key*)为：

  $h(key)=key \bmod p$　　(mod为求余运算，$p \le m$)

◆ *p*最好是质数（素数）。

3. 数字分析法

- 提取关键字中取值较均匀的数字位作为哈希地址的方法。
- 适合于所有关键字值都已知的情况，并需要对关键字中每一位的取值分布情况进行分析。

位序	1	2	3	4	5	6	7	8
	9	2	3	1	7	6	0	2
	9	2	3	2	6	8	7	5
	9	2	7	3	9	6	2	8
	9	2	3	4	3	6	3	4
	9	2	7	0	6	8	1	6
	9	2	7	7	4	6	3	8
	9	2	3	8	1	2	6	2
	9	2	3	9	4	2	2	0

哈希地址的集合为{2，75，28，34，16，38，62，20}

【示例】 假设哈希表长度m=13，采用除留余数法哈希函数建立如下关键字集合的哈希表(16，74，60，43，54，90，46，31，29，88，77)，共11个关键字。

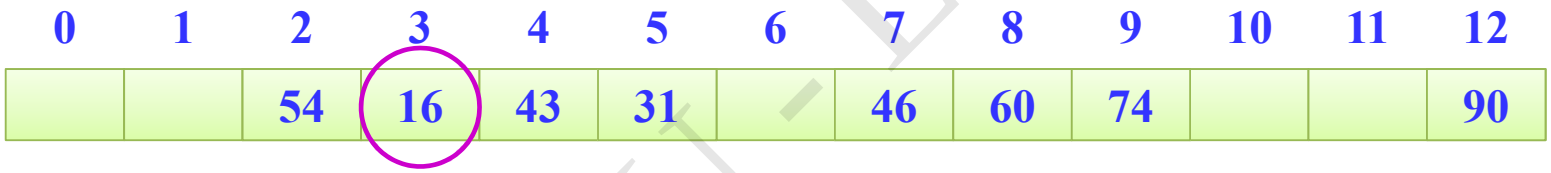
解：n=11，m=13，设计除留余数法的哈希函数为：

$$h(k)=k \bmod p$$

p应为小于等于m的素数，设p=13。

关键字：16 74 60 43 54 90 46 31 29 88 77

h(29)=3



注意：存在哈希冲突。

END