

词典

散列函数：hashCode与多项式法

有意整齐与有意变化，皆是一方死法。

于是父亲只得求助于拈阄的办法，把两个姑娘的姓氏写在两方小红纸片上，把它们揉成两团，拿在手里，走到祖宗的神主面前诚心祷告了一番，然后随意拈起一个来。李家的亲事就这样地决定了。

邓俊辉

deng@tsinghua.edu.cn

# String/Object To Integer

```
static size_t hashCode( char s[] ) {
```

```
    unsigned int h = 0;
```

```
    for ( size_t n = strlen(s), i = 0; i < n; i++ )
```

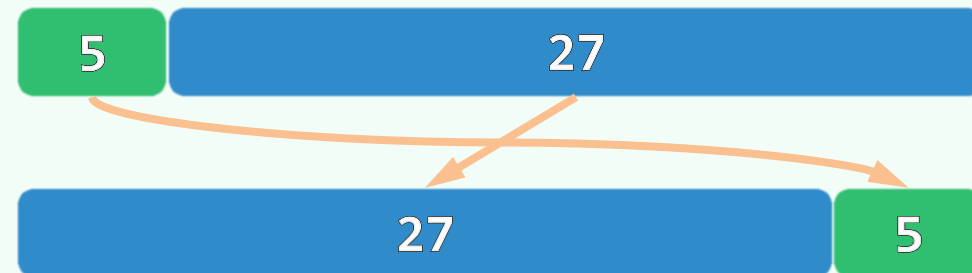
```
        { h = (h << 5) | (h >> 27); h += ( int ) s[i]; }
```

```
    return ( size_t ) h;
```

```
}
```

❖ 有必要如此复杂吗?

能否使用更简单的散列，比如...



$$\text{hashCode}("x_{n-1} \dots x_3 x_2 x_1 x_0")$$

$$= x_{n-1} \cdot a^{n-1} + \dots + x_2 \cdot a^2 + x_1 \cdot a^1 + x_0$$

$$= (\dots ((x_{n-1} \cdot a + x_{n-2}) \cdot a) + \dots + x_1) \cdot a + x_0$$

# 冲突 ~ 巧合

❖ 比如:  $hashCode(S) = \sum_{c \in S} code(upper(c))$   
 $hashCode("hash") = 8 + 1 + 19 + 8 = 36$

❖ 字符相对次序信息丢失, 将引发大量冲突

- I am Lord Voldemort
- Tom Marvolo Riddle

❖ 即便字符不同、数目不等...

- He's Harry Potter

❖ Key to improving your programming skills

Learning Tsinghua Data Structures & Algorithms

