词典

散列函数：更多

11-B2

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

（伪）随机数法

- 循环： rand(x + 1) = [a × rand(x)] % M　　//M素数，a % M ≠ 0

  a = 7^5 = 16,807 = $\boxed{100000110100111}_b$

  M = $2^{31} - 1$ = 2,147,483,647 = 01111111 $\boxed{11111111}$ 11111111 $\boxed{11111111}_b$

- 径取：hash(key) = rand(key) = [rand(0) × a^{key}] % M

  种子：rand(0) = ?

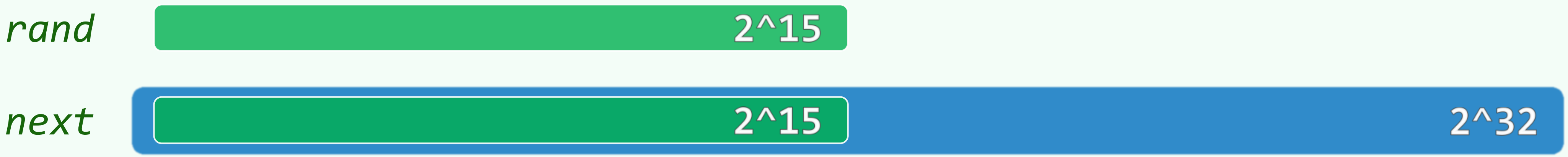
- 把难题推给伪随机数发生器，但是...

- （伪）随机数发生器的实现，因具体平台、不同历史版本而异

  创建的散列表**可移植性**差——故需慎用此法！

（伪）随机数法

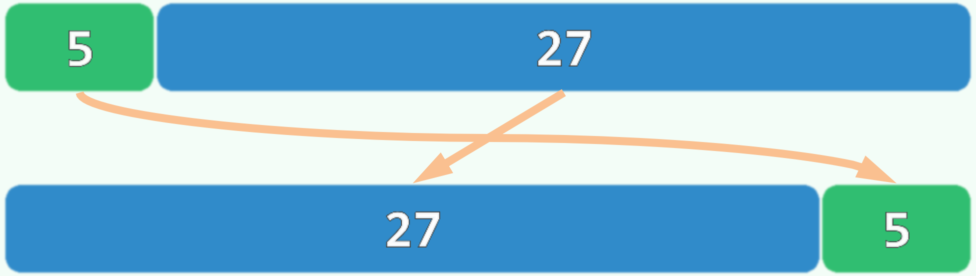
-
  ```
  unsigned long int next = 1; //The C Programming Language (2nd edn), p46
  void srand(unsigned int seed) {  next = seed;  }
  int rand(void) { //1103515245  =  3^5 * 5 * 7 * 129749
     next = next * 1103515245 + 12345;
     return (unsigned int)(next/65536) % 32768;
  }
  ```



-
  ```
  int rand() { int uninitialized; return uninitialized; }
  char* rand( t_size n ) { return ( char* ) malloc( n ); }
  ```

多项式法

```
❖ 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;
  }
```



$$hashCode(“x_{n-1} \ldots x_3 x_2 x_1 x_0”)$$

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

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

❖ 有必要如此复杂吗？

能否使用更简单的散列，比如...

冲突 ~ 巧合

- 比如： $hashCode(S) \; = \; \sum_{c \in S} code(upper(c))$

  $hashCode(\text{“}hash\text{”}) \; = \; 8+1+19+8 \; = \; 36$

- 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