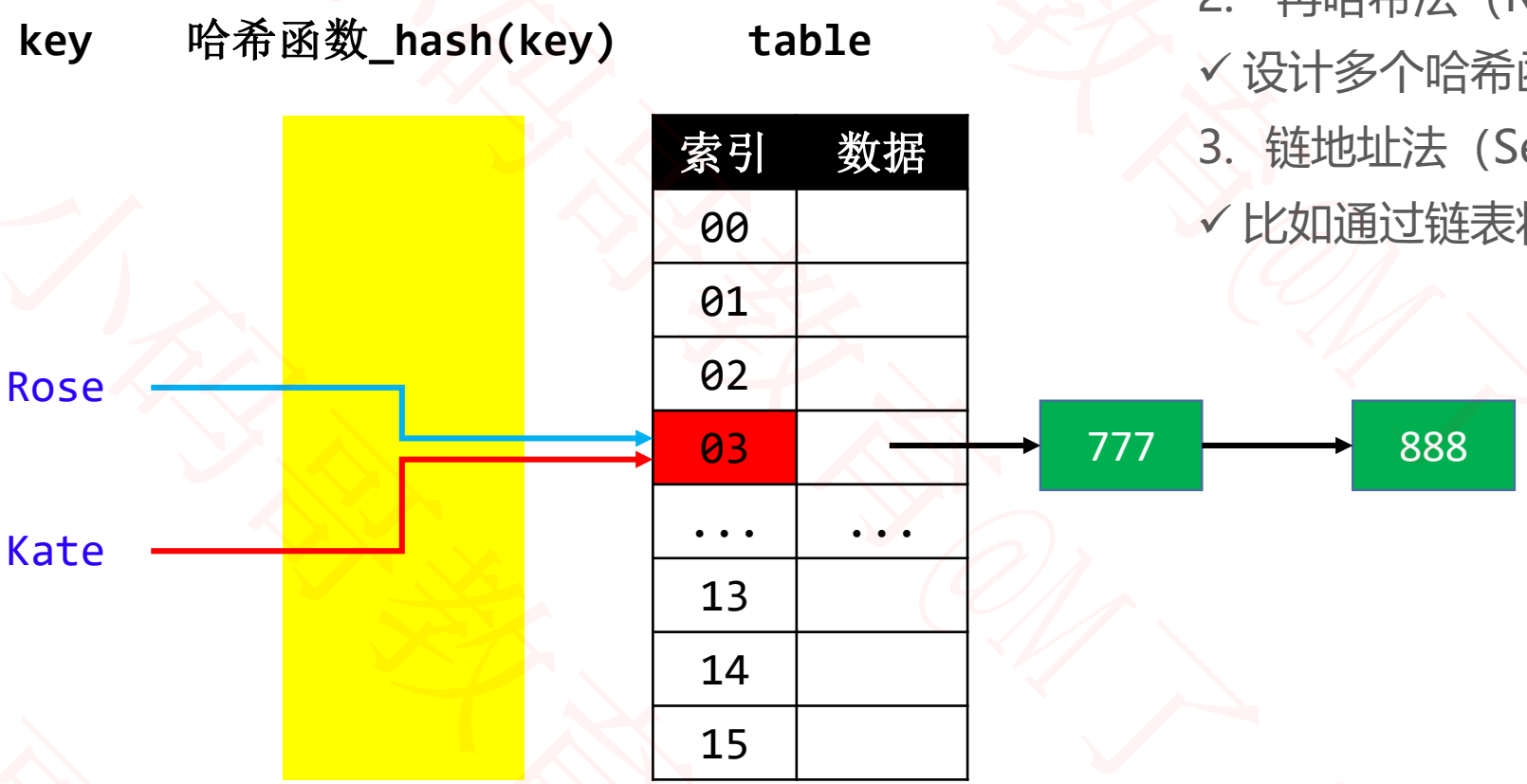
哈希冲突（Hash Collision）

- 哈希冲突也叫做哈希碰撞
  - 2 个不同的 key，经过哈希函数计算出相同的结果
  - key1 ≠ key2 ，hash(key1) = hash(key2)



- 解决哈希冲突的常见方法

1. 开放定址法（Open Addressing）

✓ 按照一定规则向其他地址探测，直到遇到空桶

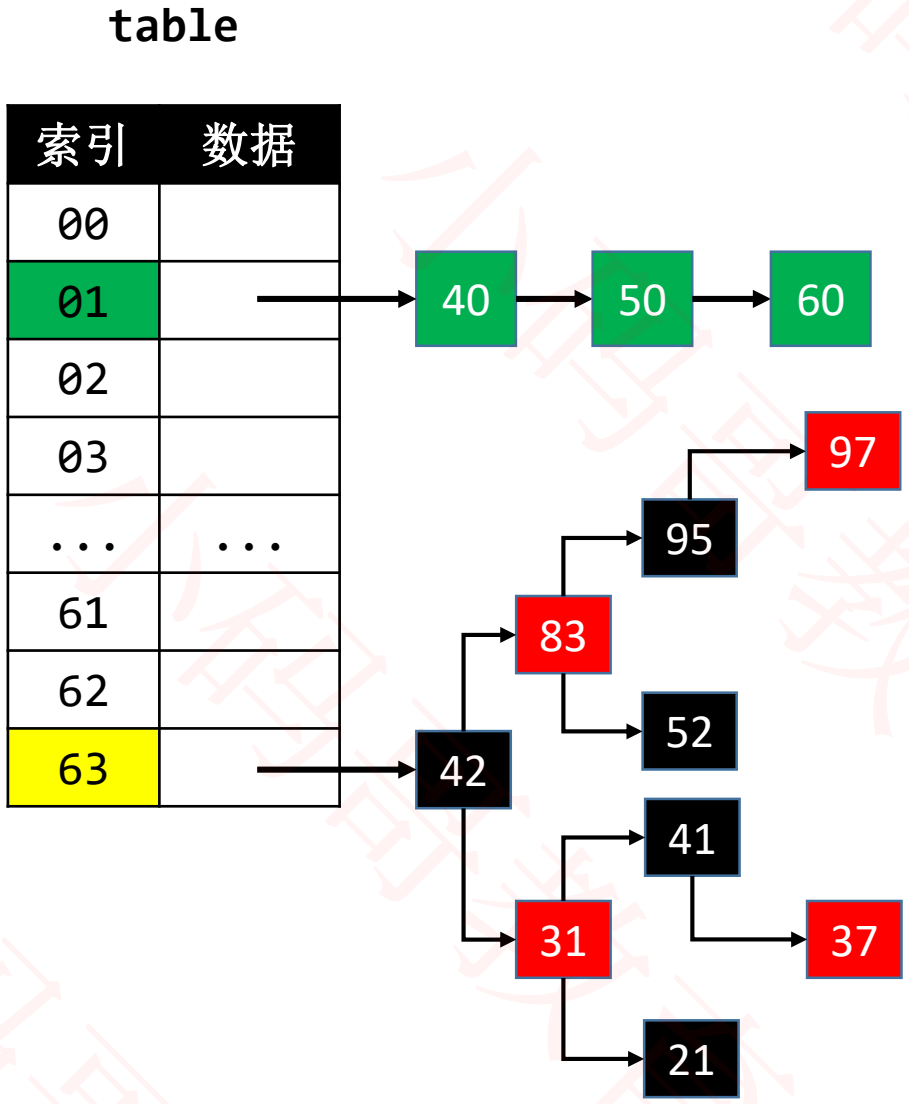
2. 再哈希法（Re-Hashing）

✓ 设计多个哈希函数

3. 链地址法（Separate Chaining）

✓ 比如通过链表将同一index的元素串起来

JDK1.8的哈希冲突解决方案



- 默认使用单向链表将元素串起来
- 在添加元素时，可能会由单向链表转为红黑树来存储元素
  - 比如当哈希表容量 ≥ 64 且 单向链表的节点数量大于 8 时
- 当红黑树节点数量少到一定程度时，又会转为单向链表
- JDK1.8中的哈希表是使用链表+红黑树解决哈希冲突
- 思考：这里为什么使用单链表?
  - 每次都是从头节点开始遍历
  - 单向链表比双向链表少一个指针，可以节省内存空间