数组：array

prepend: O(1)

append: O(1)

look up: O(1)

insert: O(n)

delete: O(n)

链表：linked list

prepend: O(1)

append: O(1)

look up: O(n)

insert: O(1)

delete: O(1)

跳表：skip list

两个重要思想：

升维

空间换时间

k级索引个数：n/(2^k)

最高级索引有2个节点：

n/(2^h) = 2

所以 h = log2(n)-1

索引高度 logn 每层索引遍历节点个数为3(最多个数）（个数可为1，2，3）

***时******间复杂度***：O(logn)

现实状态下，增删元素，索引需更新，更改索引时间复杂度为O(logn)

***空间复杂度***：O(n)

每两个抽一个：

n/2 + n/4 + n/8 ... → O(n)

每三个抽同理

LRU cache 用 linked list

redis 用 skip list