

# 算法与数据结构

## 数据结构

### 一维

#### 基础

- 数组 (array, string)
  - 泛型 (元素可以为任何数据)
  - 查找 $O(1)$
  - 插入, 删除 $O(n)$
- 链表 (linked list)
  - 单链表, 双向链表, 循环链表
  - 元素 (Node, Node 中的数据可以通过类扩展)
  - 插入, 删除  $O(1)$
  - 查找 $O(n)$
- 跳表 (skip list)
  - 有序链表
  - 对标平衡树和二分查找
  - 插入, 删除, 搜索  $O(\log n)$
  - 实现: 升维, 添加多级索引
  - 空间复杂度 $O(n)$
  - 应用: LRU Cache, Redis

#### 高级

- 栈 (stack)
  - 添加, 删除  $O(1)$
  - 查找  $O(n)$
  - 无序即为 N, 遍历
- 队列 (queue)
  - 添加, 删除  $O(1)$
  - 查找  $O(n)$
- 双端队列 (deque)
  - 添加, 删除  $O(1)$
  - 查找  $O(n)$
- 优先队列 (priority queue)
  - 取出  $O(\log N)$
  - 底层实现多样性 (heap, bst, treap)
- 集合 (set)
- 映射 (map, hash)

### 二维

#### 基础

- 树 (tree)
- 图 (graph)
- 二叉搜索树 (binary search tree, red-black tree, AVL)

#### 高级

- 堆 (heap)
- 并查集 (disjoint set)
- 字典树 (trie)

### 特殊

- 位运算 (bitwise)
- 布隆过滤器 (bloom filter)
- LRU Cache

## 算法

### 常用方法

- 一二三... 重遍历 (枚举)
- 开辟新的数据空间, 逐个复制
- 嵌套循环
- 快慢指针
- 左右夹逼
- 中心扩散
- 字典
- hash
- 一层遍历 + 索引检索
- 最近重复子问题
- 递归
- 最近相关性
- 栈
- 单调栈 -- 查找边界

#### 基础

- if-else, switch --> branch
- for, while loop --> iteration
- 递归 Recursion (Divide & Conquer, Backtrace)

#### 高级

- 搜索 search
  - 深度优先搜索 Depth first search (DFS)
  - 广度优先搜索 Breadth first search (BFS)
- 动态规划 Dynamic Programming
- 二分查找 Binary Search
- 贪心 Greedy
- 数学 Math
- 几何 Geometry

#### 时间复杂度

- $O(1)$
- $O(\log n)$
- $O(n)$
- $O(n^2)$
- $O(n^3)$