

数据结构

Data Structure

AHdoc



Recall

- 队列，栈，链表（双向链表）
- **HASH**
- 堆
- 排序二叉树，平衡树
- 并查集



今日话题

- 1，特殊的数据结构
- 2，线段树
- 3，线段树 与 树



1，特殊的数据结构

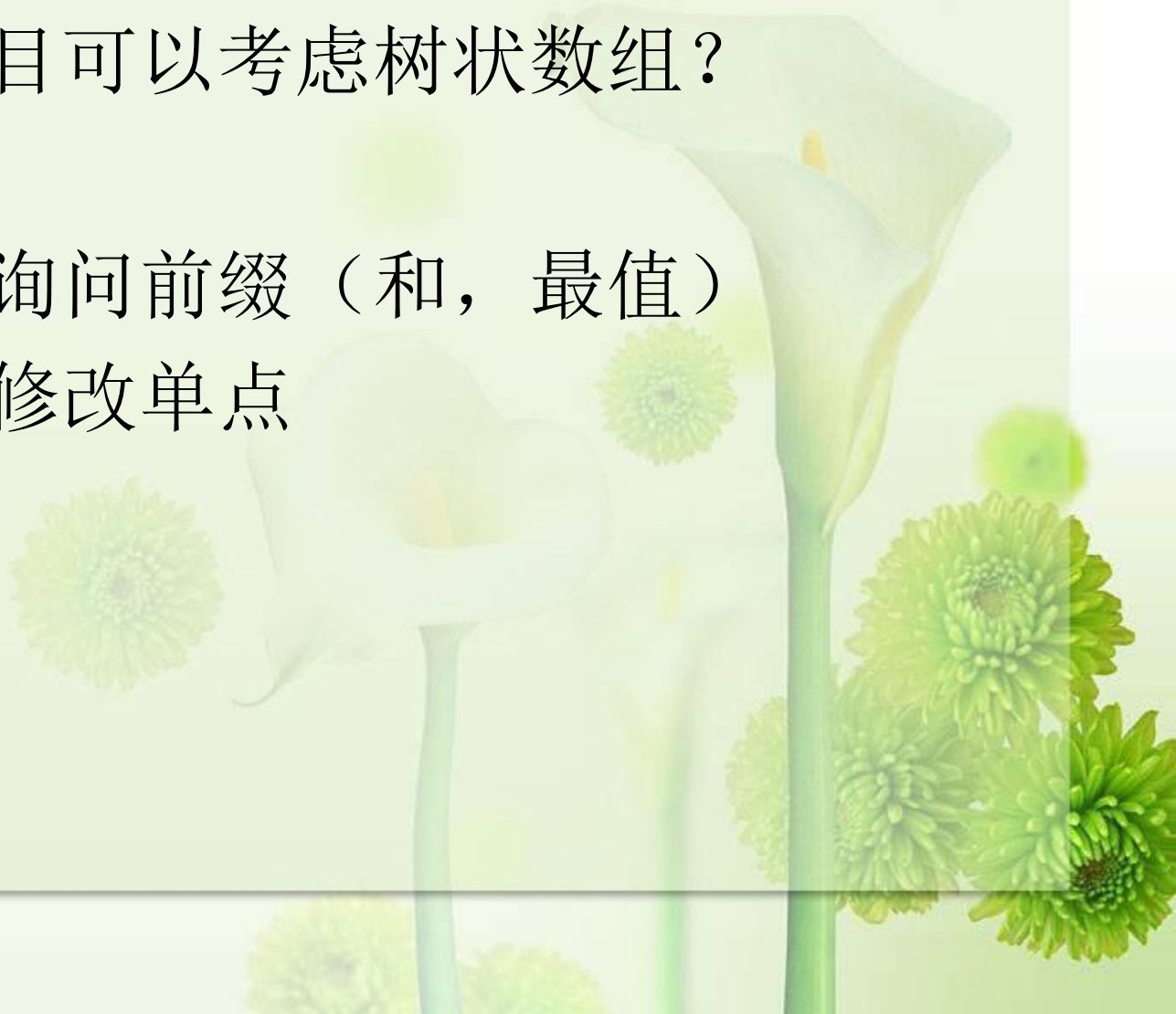
- 1.1，树状数组
- 1.2，稀疏表（**ST**）
- 1.3，树上倍增



1，特殊的数据结构

1.1，树状数组

- 什么样的题目可以考虑树状数组？
- 询问形式：询问前缀（和，最值）
- 维护修改：修改单点



1, 特殊的数据结构

1.1, 树状数组

- `void add(int x, int c) {`
- `for (;x <= N; x += (x & (-x))) Sum[x] += c;`
- `}`

- `void FindSum(int x) {`
- `int ret = 0;`
- `for (;x >= 1; x -= (x & (-x))) ret += Sum[x];`
- `return ret;`
- `}`

1, 特殊的数据结构

1.2, Sparse Table (ST)

- 支持快速询问：
 - 区间和
 - 区间最值
 - 可通过“2个区间信息合并”得到的任何形式的信息
- 不支持修改
- 预处理 $O(n \lg n)$
- 询问 $O(\lg n) - O(1)$
- 代码量 极短

1, 特殊的数据结构

1.2, Sparse Table (ST)

- 用ST轻松实现离线LCA
- 树结构到线序结构? DFS序!
- 权值: 深度h
- 求区间最小值 (为什么)
- 合并两个区间
 - $F[i][2^j] = \min\{F[i][2^{j-1}], F[i+2^{j-1}][j-1]\}$

1, 特殊的数据结构

1.3, 树上倍增

- 与ST的相似性：都是基于动态规划思想
- $F[i][2^j]$: 从 i 出发向上走 2^j 距离

1, 特殊的数据结构

1.3, 树上倍增

- Query on a tree II
- You are given a tree (an undirected acyclic connected graph) with N nodes, and edges numbered $1, 2, 3 \dots N-1$. Each edge has an integer value assigned to it, representing its length.
- We will ask you to perform some instructions of the following form:
 - DIST $a\ b$: ask for the distance between node a and node b
 - KTH $a\ b\ k$: ask for the k -th node on the path from node a to node b

2, 线段树

- 维护什么？
 - 维护缺少什么
- 三个重点：
 - 在一个区间上（对应线段树一个结点）的多次区间操作，可以合并为同一个区间操作。
 - 一个区间上的操作可以分解为对左右2个区间的操作。
 - 可以快速合并两个区间的信息。

2, 线段树一例1

有一个平面连轴支架，包含 n 根棍子，相邻两根有一个连接点。第一根有一端（不连第二根的那端）固定在 $(0, 0)$ 。维护三个操作：

- (1) 绕某一连接点旋转（这个点把所有棍子分成两部分，他们旋转时分别是一个整体）
- (2) 某根棍子拉长或缩短
- (3) 询问某连接点坐标

2, 线段树一例2

— 输入一个长为 n 的数列，维护 m 个操作，操作分为三类：

- (1)某连续段一起加上一个常数
- (2)询问某一段的所有数的两两乘积的和
- (3)询问某一段的所有相邻两数乘积的和

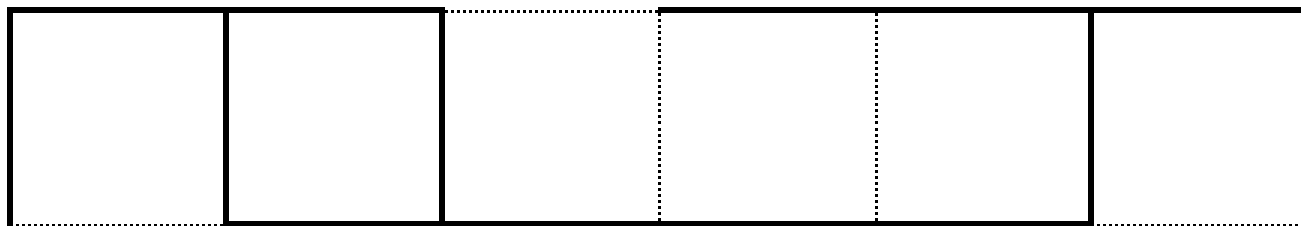
2, 线段树一例3

- 输入长度为 n 的数列 $a[i]$ 。维护 m 次操作，每次操作可以：
 - (1) $a[l] \dots a[r]$ 每一项都加一个常数 C
 - (2)求 $F[a[l]] + F[a[l]+1] + \dots + F[a[r]] \bmod 10001$ 的余数
 - (3)求 $F[a[l]] + F[a[l+1]] + \dots + F[a[r]] \bmod 10001$ 的余数
- 其中 $F[i]$ 表示斐波那契数列。即 $F[0]=F[1]=1$,
 $F[n+2]=F[n+1]+F[n]$ 。 ($C \leq 10^{11}$)

2, 线段树一例4

- 有一个 $2*n$ 的点阵，平行于坐标轴的方向上相邻的点之间可以连边，维护以下操作：
 - (1)在某两点之间连边（可以连边的话）
 - (2)拆除某条边
 - (3)询问某两点是否连通

2, 线段树一例4



3， 线段树 与 树

- 将树拆分为链
 - 思维复杂度“高”：DFS序
 - 代码复杂度高：树链剖分
- 思维角度：
 - 首先思考链上的情况
 - 再思考多个链之间的拼接情况

树链剖分

- 1) 如何把树拆分为若干链
- 2) 如何考虑两点之间的路径
- 3) 如何保证高效性
- 4) 如何维护这些链
- 5) 如何考察子树



3, 线段树 与 树—例5

- 输入一个 n 个点的边有权的有根树。维护 m 个操作，操作共有如下两类：
 - (1)将某子树中的所有边权都增加一个常数 C
 - (2)求某两点之间路径上的边权的平方和

3, 线段树 与 树—例6

• Minimum Cut-Cut

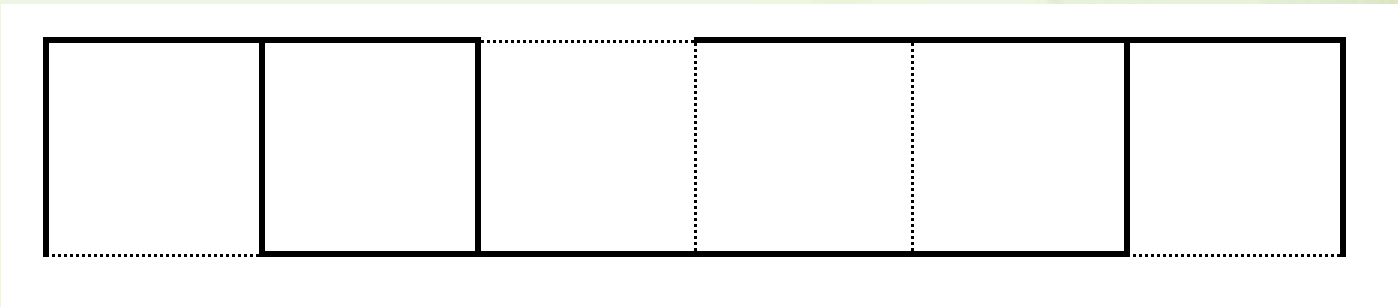
- Given a simple unweighted graph G (an undirected graph without self-loops or multiple edges) with n nodes and m edges. Let T be a spanning tree of G . We say that a cut in G two-respects T if and only if it cuts **just two edges** of T .
- You should find the minimum cut of graph G two-respecting given spanning tree T .
- To simplify the problem, we guarantee that for each edge (u,v) in T in graph G , the unique path in T between u and v must pass through the node 1.

3, 线段树 与 树—例6

- Step 1: 以1为根建立有根树
 - Step 2: 树链剖分
 - Step 3: 找一条极小的链, L
 - Step 4: 沿着链询问, 修改
 - Step 5: 把所有这条链上的回边都挪动到顶端
 - Step 6: 删除 L , 回Step 3
-
- 每一条边被考察（并挪动）了 $O(\lg N)$ 次

3, 线段树 与 树—例7

- 回顾:



- 很粗的树
 - 树 T —图 G
 - 树上的结点 s —图上若干个结点
 - 树上的边— s_i 与 s_j 中公共的点

3, 线段树 与 树—例7

- 很粗的树
 - 树 T —图 G
 - 有多粗?
 - $w := T$ 中每一个点包含的 G 中点数的最大值—1
- 树链剖分后的链
 - 对应到一维看是什么?
 - 如何维护?
 - 维护的复杂度?
- 整个结构上的询问:
 - 最短路径
 - 其它?

今日训练题

- <http://acm.hust.edu.cn/vjudge/contest/125873>
- Problem A. Can you answer these queries I
- Problem B. Frequent values
- Problem C. 维护序列
- Problem D. Apple Tree
- Problem E. Counting Offspring
- Problem F. Adam and Tree
- 欢迎提问！