

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

# 数据结构相关

戴傅聪

# 目录

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

1 倍增

2 求 LCA 的几种算法

3 STL

4 二叉堆

5 二叉搜索树

6 线段树、树状数组

# 倍增

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 由于 st 表特殊的应用比较少，我们这里就直接讲倍增了

# 倍增

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 由于 st 表特殊的应用比较少，我们这里就直接讲倍增了
- 倍增不支持修改，通常遇到带修改的题目则需要使用其他数据结构代替

# st 表

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- $O(n \log n)$  预处理后  $O(1)$  求区间最大值

# st 表

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- $O(n \log n)$  预处理后  $O(1)$  求区间最大值
- 令  $f_{i,j}$  表示  $[i, i + 2^j - 1]$  范围内的最大值

# st 表

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- $O(n \log n)$  预处理后  $O(1)$  求区间最大值
- 令  $f_{i,j}$  表示  $[i, i + 2^j - 1]$  范围内的最大值
- 则有  $f_{i,j} = \max(f_{i,j-1}, f_{i+2^{j-1},j-1})$

# st 表

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- $O(n \log n)$  预处理后  $O(1)$  求区间最大值
- 令  $f_{i,j}$  表示  $[i, i + 2^j - 1]$  范围内的最大值
- 则有  $f_{i,j} = \max(f_{i,j-1}, f_{i+2^{j-1},j-1})$
- 询问时令  $t = \log_2(r - l + 1)$ , 则  $\max(f_{l,t}, f_{r-2^t+1,t})$  即为答案, 因为  $[l, l + 2^t - 1], [r - 2^t + 1, r]$  这两个区间完全覆盖了  $[l, r]$



# NOIP2012 开车旅行

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 题意较为繁琐，丢个链接

# NOIP2012 开车旅行

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 题意较为繁琐，丢个链接
- <https://www.luogu.com.cn/problem/P1081>

# NOIP2012 开车旅行

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 首先预处理出二人从所有位置开车会开到哪里。这部分可以用链表  $O(n)$  解决。

# NOIP2012 开车旅行

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 首先预处理出二人从所有位置开车会开到哪里。这部分可以用链表  $O(n)$  解决。
- 于是接下来就要对于确定的路程限制  $x$  和出发城市，求出两个人分别的前进距离。

# NOIP2012 开车旅行

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 首先预处理出二人从所有位置开车会开到哪里。这部分可以用链表  $O(n)$  解决。
- 于是接下来就要对于确定的路程限制  $x$  和出发城市，求出两个人分别的前进距离。
- 我们令  $f_{i,j}, g_{i,j}$  分别表示从  $i$  出发，二人各开  $2^j$  次，A 开的路程与 B 开的路程。

# NOIP2012 开车旅行

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 首先预处理出二人从所有位置开车会开到哪里。这部分可以用链表  $O(n)$  解决。
- 于是接下来就要对于确定的路程限制  $x$  和出发城市，求出两个人分别的前进距离。
- 我们令  $f_{i,j}, g_{i,j}$  分别表示从  $i$  出发，二人各开  $2^j$  次，A 开的路程与 B 开的路程。
- 然后令  $j$  从  $\log n$  循环到 0，如果二人开  $2^j$  轮后不会超出路程限制，则让两个人到达开  $2^j$  轮后的位置并加上路程即可。注意最后需要判断一下 A 能不能再开一次

# JOISC2021 活动参观

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 给定  $n$  个区间，要求从中选出互不相交的  $k$  个，且选出的区间标号字典序最小。

# JOISC2021 活动参观

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 给定  $n$  个区间，要求从中选出互不相交的  $k$  个，且选出的区间标号字典序最小。
- $n \leq 10^5$



# JOISC2021 活动参观

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 选出最多的区间是一个经典的贪心，将所有区间按照右端点从小到大排序，然后碰到能选的就选即可。

# JOISC2021 活动参观

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 选出最多的区间是一个经典的贪心，将所有区间按照右端点从小到大排序，然后碰到能选的就选即可。
- 为了最小化字典序。我们按照区间的标号从小到大尝试选区间，然后求出强制选这个区间后，剩下的部分能不能选够  $k$  个。

# JOISC2021 活动参观

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 选出最多的区间是一个经典的贪心，将所有区间按照右端点从小到大排序，然后碰到能选的就选即可。
- 为了最小化字典序。我们按照区间的标号从小到大尝试选区间，然后求出强制选这个区间后，剩下的部分能不能选够  $k$  个。
- 需要判断没有被覆盖的区间内最多能选出几个，区间与区间之间是独立的，且每次强制选一个区间只会影响一个区间。

# JOISC2021 活动参观

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 选出最多的区间是一个经典的贪心，将所有区间按照右端点从小到大排序，然后碰到能选的就选即可。
- 为了最小化字典序。我们按照区间的标号从小到大尝试选区间，然后求出强制选这个区间后，剩下的部分能不能选够  $k$  个。
- 需要判断没有被覆盖的区间内最多能选出几个，区间与区间之间是独立的，且每次强制选一个区间只会影响一个区间。
- 于是接下来问题就变成了，多次询问  $[l, r]$  范围内最多选出多少个区间。

# JOISC2021 活动参观

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 我们需要在前面的基础的贪心上优化。我们需要求出  $f_{i,j}$  表示选出第  $i$  个区间后，往后选  $2^j$  个区间会选到哪个。

# JOISC2021 活动参观

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 我们需要在前面的基础的贪心上优化。我们需要求出  $f_{i,j}$  表示选出第  $i$  个区间后，往后选  $2^j$  个区间会选到哪个。
- 然后我们找到  $[l, r]$  内右端点最小的区间作为起点开始倍增即可。

# JOISC2021 活动参观

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 我们需要在前面的基础的贪心上优化。我们需要求出  $f_{i,j}$  表示选出第  $i$  个区间后，往后选  $2^j$  个区间会选到哪个。
- 然后我们找到  $[l, r]$  内右端点最小的区间作为起点开始倍增即可。
- 总时间复杂度  $O(n \log n)$

# 目录

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

1 倍增

2 求 LCA 的几种算法

3 STL

4 二叉堆

5 二叉搜索树

6 线段树、树状数组



# 倍增求 LCA

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 题意略

# 倍增求 LCA

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 令  $f_{i,j}$  节点  $i$  的  $2^j$  级祖先。

# 倍增求 LCA

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 令  $f_{i,j}$  节点  $i$  的  $2^j$  级祖先。
- 询问两个点的 LCA 时，首先将深度较深的节点跑到和另一个节点相同深度。这部分可以求出深度差的二进制表示然后利用  $f$  来完成。如果到同一深度后两个节点重合，那么这个重合的节点就是答案。

# 倍增求 LCA

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 令  $f_{i,j}$  节点  $i$  的  $2^j$  级祖先。
- 询问两个点的 LCA 时，首先将深度较深的节点跑到和另一个节点相同深度。这部分可以求出深度差的二进制表示然后利用  $f$  来完成。如果到同一深度后两个节点重合，那么这个重合的节点就是答案。
- 否则我们按照  $j$  从大到小的顺序依次尝试将两个节点同时往上爬  $2^j$  个节点，如果两个节点仍然不重合就呆在  $2^j$  级父亲处，否则就呆在原地。

# 倍增求 LCA

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 令  $f_{i,j}$  节点  $i$  的  $2^j$  级祖先。
- 询问两个点的 LCA 时，首先将深度较深的节点跑到和另一个节点相同深度。这部分可以求出深度差的二进制表示然后利用  $f$  来完成。如果到同一深度后两个节点重合，那么这个重合的节点就是答案。
- 否则我们按照  $j$  从大到小的顺序依次尝试将两个节点同时往上爬  $2^j$  个节点，如果两个节点仍然不重合就呆在  $2^j$  级父亲处，否则就呆在原地。
- 最后两个节点的父亲一定是相同的，这个父亲节点就是答案。

# 倍增求 LCA

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 令  $f_{i,j}$  节点  $i$  的  $2^j$  级祖先。

# 倍增求 LCA

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 令  $f_{i,j}$  节点  $i$  的  $2^j$  级祖先。
- 询问两个点的 LCA 时，首先将深度较深的节点跑到和另一个节点相同深度。这部分可以求出深度差的二进制表示然后利用  $f$  来完成。如果到同一深度后两个节点重合，那么这个重合的节点就是答案。

# 倍增求 LCA

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 令  $f_{i,j}$  节点  $i$  的  $2^j$  级祖先。
- 询问两个点的 LCA 时，首先将深度较深的节点跑到和另一个节点相同深度。这部分可以求出深度差的二进制表示然后利用  $f$  来完成。如果到同一深度后两个节点重合，那么这个重合的节点就是答案。
- 否则我们按照  $j$  从大到小的顺序依次尝试将两个节点同时往上爬  $2^j$  个节点，如果两个节点仍然不重合就呆在  $2^j$  级父亲处，否则就呆在原地。



# 倍增求 LCA

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 令  $f_{i,j}$  节点  $i$  的  $2^j$  级祖先。
- 询问两个点的 LCA 时，首先将深度较深的节点跑到和另一个节点相同深度。这部分可以求出深度差的二进制表示然后利用  $f$  来完成。如果到同一深度后两个节点重合，那么这个重合的节点就是答案。
- 否则我们按照  $j$  从大到小的顺序依次尝试将两个节点同时往上爬  $2^j$  个节点，如果两个节点仍然不重合就呆在  $2^j$  级父亲处，否则就呆在原地。
- 最后两个节点的父亲一定是相同的，这个父亲节点就是答案。

# RMQ 求 LCA

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 首先我们求出这棵树的欧拉序，欧拉序是在 dfs 过程中，每次 dfs 到一个节点时先将这个节点 **pushback** 到序列中，然后每 dfs 完这个节点的一棵子树，也要将这个节点 **pushback** 到序列中。

# RMQ 求 LCA

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 首先我们求出这棵树的欧拉序，欧拉序是在 dfs 过程中，每次 dfs 到一个节点时先将这个节点 `pushback` 到序列中，然后每 dfs 完这个节点的一棵子树，也要将这个节点 `pushback` 到序列中。
- 经过细致的观察，我们可以发现：两个节点的 LCA 是这两个节点在欧拉序的区间中（一个节点可能在欧拉序中出现多次，区间端点取任意一次出现均可），深度最小的节点。

# RMQ 求 LCA

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 首先我们求出这棵树的欧拉序，欧拉序是在 dfs 过程中，每次 dfs 到一个节点时先将这个节点 `pushback` 到序列中，然后每 dfs 完这个节点的一棵子树，也要将这个节点 `pushback` 到序列中。
- 经过细致的观察，我们可以发现：两个节点的 LCA 是这两个节点在欧拉序的区间中（一个节点可能在欧拉序中出现多次，区间端点取任意一次出现均可），深度最小的节点。
- 因此我们可以求出这棵树的欧拉序，然后使用 `st` 表预处理即可  $O(1)$  求 LCA。

# tarjan 求 LCA

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 离线算法，除了一些特别卡的题目以外基本上没什么用。

# tarjan 求 LCA

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 离线算法，除了一些特别卡的题目以外基本上没什么用。
- 使用并查集将已经结束 **dfs** 的部分向父亲合并，然后在每组询问后访问到的节点处查询先访问到的节点被合并到哪里了。

# 树剖求 LCA

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 每个节点的重儿子是这个节点的所有儿子中子树 size 最大的那个。

# 树剖求 LCA

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 每个节点的重儿子是这个节点的所有儿子中子树 size 最大的那个。
- 这样一个节点到根的路径上最多  $O(\log n)$  条轻边。



# 树剖求 LCA

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 每个节点的重儿子是这个节点的所有儿子中子树 size 最大的那个。
- 这样一个节点到根的路径上最多  $O(\log n)$  条轻边。
- 求两个点的 lca 时，每次不断将链顶端深度较大（注意是链顶端的深度，而不是节点的深度）的点跑到链顶端的父亲，直到两个点在同一条众链上即可。

# 目录

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

1 倍增

2 求 LCA 的几种算法

3 STL

4 二叉堆

5 二叉搜索树

6 线段树、树状数组

# STL 函数

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 常用的函数有  
`sort`, `stable_sort`, `lower_bound`, `upper_bound`, `unique`, `make_heap`  
等。

# STL 函数

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 常用的函数有 `sort`, `stable_sort`, `lower_bound`, `upper_bound`, `unique`, `make_heap` 等。
- `stable_sort` 为具有稳定性的排序，用法与 `sort` 相同，均为前两个参数指定左闭右开的区间，第三个参数可以传比较函数或大于号（"`greater<int>()`"）。

# STL 函数

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 常用的函数有 `sort`, `stable_sort`, `lower_bound`, `upper_bound`, `unique`, `make_heap` 等。
- `stable_sort` 为具有稳定性的排序，用法与 `sort` 相同，均为前两个参数指定左闭右开的区间，第三个参数可以传比较函数或大于号（"`greater<int>()`"）。
- `unique` 有两个参数表示左闭右开的区间，返回值是最后一个元素的后面一个位置的指针。

# STL 函数

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 常用的函数有 `sort`, `stable_sort`, `lower_bound`, `upper_bound`, `unique`, `make_heap` 等。
- `stable_sort` 为具有稳定性的排序，用法与 `sort` 相同，均为前两个参数指定左闭右开的区间，第三个参数可以传比较函数或大于号（"`greater<int>()`"）。
- `unique` 有两个参数表示左闭右开的区间，返回值是最后一个元素的后面一个位置的指针。
- 上面两个是最常用的函数，`unique` 的主要应用在于离散化

# STL 函数

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 常用的函数有 `sort`, `stable_sort`, `lower_bound`, `upper_bound`, `unique`, `make_heap` 等。
- `stable_sort` 为具有稳定性的排序，用法与 `sort` 相同，均为前两个参数指定左闭右开的区间，第三个参数可以传比较函数或大于号（"`greater<int>()`"）。
- `unique` 有两个参数表示左闭右开的区间，返回值是最后一个元素的后面一个位置的指针。
- 上面两个是最常用的函数，`unique` 的主要应用在于离散化
- 其他函数的用法可以自行搜索

# STL 容器

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 常用的容器有  
deque, vector, stack, queue, priority\_queue, set, map, multiset, multimap  
等等



# STL 容器

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 常用的容器有  
deque, vector, stack, queue, priority\_queue, set, map, multiset, multimap 等等
- 注意部分容器即使为空也会占据一定量的内存，尤其是  
deque，因此在空间较紧张时不要开大量的 deque

# STL 容器

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 常用的容器有  
deque,vector,stack,queue,priority\_queue,set,map,multiset,multimap 等等
- 注意部分容器即使为空也会占据一定量的内存，尤其是 deque，因此在空间较紧张时不要开大量的 deque
- 其中 deque,vector 支持随机访问（像数组一样访问），stack,queue,priority\_queue 仅支持访问栈顶或队头元素且不支持迭代器，set,map,multiset,multimap 仅支持顺序访问（map 的[]功能视为容器的查找功能）

# STL 容器

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 大部分容器都有自己的迭代器，例如 `deque` 的迭代器类型为 `deque::iterator`，其中支持随机访问的容器的迭代器是随机访问迭代器，支持 `++`, `--`, 加减一个整数, 比较大小的操作，可以用 `*` 来得到迭代器指向的内容。支持顺序访问的容器的迭代器仅支持 `++`, `--` 以及比较等于/不等于。因此在 `set` 一类的容器中使用 `lower_bound` 应该用 `s.lower_bound(x)`，而不是 `lower_bound(s.begin(), s.end(), x)`，后者的时间复杂度是错误的。

# STL 容器

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 大部分容器都有自己的迭代器，例如 `deque` 的迭代器类型为 `deque::iterator`，其中支持随机访问的容器的迭代器是随机访问迭代器，支持 `++`, `--`, 加减一个整数, 比较大小的操作，可以用 `*` 来得到迭代器指向的内容。支持顺序访问的容器的迭代器仅支持 `++`, `--` 以及比较等于/不等于。因此在 `set` 一类的容器中使用 `lower_bound` 应该用 `s.lower_bound(x)`，而不是 `lower_bound(s.begin(), s.end(), x)`，后者的时间复杂度是错误的。
- 使用 `.begin()` 和 `.end()` 可以得到容器的开头与结尾的迭代器，注意这里仍然是左闭右开的。

# STL 容器

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- deque 随机访问、在开头结尾加入元素的复杂度均为  $O(1)$

# STL 容器

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- **deque** 随机访问、在开头结尾加入元素的复杂度均为  $O(1)$
- **vector** 随机访问，在结尾加入元素的复杂度为  $O(1)$ ，但在开头或中间插入元素无法保证复杂度。

# STL 容器

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- deque 随机访问、在开头结尾加入元素的复杂度均为  $O(1)$
- vector 随机访问，在结尾加入元素的复杂度为  $O(1)$ ，但在开头或中间插入元素无法保证复杂度。
- set 与 map 内部通过平衡树实现，因此常见的操作均为  $\log$  级别

# STL 容器

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- deque 随机访问、在开头结尾加入元素的复杂度均为  $O(1)$
- vector 随机访问，在结尾加入元素的复杂度为  $O(1)$ ，但在开头或中间插入元素无法保证复杂度。
- set 与 map 内部通过平衡树实现，因此常见的操作均为  $\log$  级别
- 需要注意的是 multiset 的 count 复杂度是  $\log n + ans$ ，其中  $ans$  是元素出现的次数。



# bitset

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- **bitset** 可以方便而高效地维护压位二进制数组。

# bitset

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- `bitset` 可以方便而高效地维护压位二进制数组。
- 内置了 `none`, `all`, `flip`, `count`, `set`, `reset` 等函数，支持左移、右移操作，复杂度均为  $O(|L|/32)$

# bitset

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- bitset 可以方便而高效地维护压位二进制数组。
- 内置了 none,all,flip,count,set,reset 等函数，支持左移、右移操作，复杂度均为  $O(|L|/32)$
- 可以快速地计算背包类 DP（数据范围可以开到  $10^5$  左右）

# 目录

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

1 倍增

2 求 LCA 的几种算法

3 STL

4 二叉堆

5 二叉搜索树

6 线段树、树状数组

# 二叉堆

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 是一个完全二叉树，树上每个节点均有一个不大于父亲的权值

# 二叉堆

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 是一个完全二叉树，树上每个节点均有一个不大于父亲的权值
- 在数组中的存储方式为： $n$  个点在数组中的下标范围为  $[1, n]$ ， $i$  的儿子是  $i * 2$  与  $i * 2 + 1$

# 二叉堆

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 是一个完全二叉树，树上每个节点均有一个不大于父亲的权值
- 在数组中的存储方式为： $n$  个点在数组中的下标范围为  $[1, n]$ ， $i$  的儿子是  $i * 2$  与  $i * 2 + 1$
- 插入一个元素时，在  $n + 1$  的位置放入元素，然后不断与父亲比较并交换直到比父亲小为止。

# 二叉堆

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 是一个完全二叉树，树上每个节点均有一个不大于父亲的权值
- 在数组中的存储方式为： $n$  个点在数组中的下标范围为  $[1, n]$ ， $i$  的儿子是  $i * 2$  与  $i * 2 + 1$
- 插入一个元素时，在  $n + 1$  的位置放入元素，然后不断与父亲比较并交换直到比父亲小为止。
- 删除堆顶元素时，将堆顶与  $n$  的位置互换，然后将  $n$  减一，将新的堆顶元素不断向儿子方向调整，使其满足堆的性质。



# 二叉堆

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 是一个完全二叉树，树上每个节点均有一个不大于父亲的权值
- 在数组中的存储方式为： $n$  个点在数组中的下标范围为  $[1, n]$ ， $i$  的儿子是  $i * 2$  与  $i * 2 + 1$
- 插入一个元素时，在  $n + 1$  的位置放入元素，然后不断与父亲比较并交换直到比父亲小为止。
- 删除堆顶元素时，将堆顶与  $n$  的位置互换，然后将  $n$  减一，将新的堆顶元素不断向儿子方向调整，使其满足堆的性质。
- 由于可以用 `priority_queue` 或 `set, map` 等更有用的 stl 代替所以并没有什么卵用。

# 对顶堆

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 给定一个长为  $n$  的数组，求所有奇数长度的前缀的中位数

# 对顶堆

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 维护两个堆，其中一个大根堆维护较小的一半元素，一个小根堆维护较大的一半元素

# 对顶堆

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 维护两个堆，其中一个大根堆维护较小的一半元素，一个小根堆维护较大的一半元素
- 加入一个数后对两个堆进行调整，使得大根堆的元素比小根堆的都要小，调整的量是  $O(1)$  的

# 目录

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

1 倍增

2 求 LCA 的几种算法

3 STL

4 二叉堆

5 二叉搜索树

6 线段树、树状数组

# 二叉搜索树

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 是一棵二叉树，每个节点处有一个权值，其左子树所有元素均小于自己的权值，右子树所有元素均大于自己的权值。

# 二叉搜索树

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 是一棵二叉树，每个节点处有一个权值，其左子树所有元素均小于自己的权值，右子树所有元素均大于自己的权值。
- 其中序遍历是一个有序的序列，可以支持插入、删除、查找元素、在二叉搜索树上二分等操作。

# 二叉搜索树的旋转

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 我们可以在不改变二叉搜索树性质的情况下少量地修改其祖先关系，使得树的形态发生变化，这是二叉搜索树的旋转。



# 二叉搜索树的旋转

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

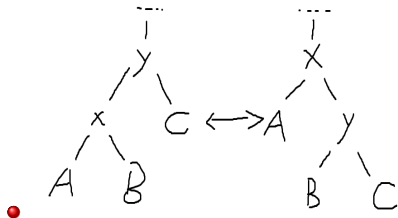
STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 我们可以在不改变二叉搜索树性质的情况下少量地修改其祖先关系，使得树的形态发生变化，这是二叉搜索树的旋转。



# 二叉搜索树的插入

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 从根节点开始，如果插入元素比当前节点小就往左子树跑，否则往右子树跑

# 二叉搜索树的插入

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 从根节点开始，如果插入元素比当前节点小就往左子树跑，否则往右子树跑
- 跑到空节点时在空节点处新建一个节点即可。

# 二叉搜索树的删除

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 把删除的节点通过不断旋转，使其成为叶子节点

# 二叉搜索树的删除

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 把删除的节点通过不断旋转，使其成为叶子节点
- 然后直接删除即可

# 二叉搜索树的查找

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 从根节点开始，如果查找的值等于当前节点的值就直接返回，如果查找的值小于当前节点的值就往左子树走，否则往右子树走。

# 二叉搜索树的查找

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 从根节点开始，如果查找的值等于当前节点的值就直接返回，如果查找的值小于当前节点的值就往左子树走，否则往右子树走。
- 走到空节点的话说明找不到

# 二叉搜索树上的二分

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 以找第  $k$  个元素为例，首先在刚才的操作中我们需要维护所有子树的大小



# 二叉搜索树上的二分

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 以找第  $k$  个元素为例，首先在刚才的操作中我们需要维护所有子树的大小
- 然后从根节点开始，如果左子树的大小大于等于  $k$  就跑左子树，刚好等于  $k-1$  那就是这个节点，如果小于  $k-1$  就跑右子树并把  $k$  减去左子树大小加一

# 平衡树

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 在之前的操作中，我们并没有关注二叉搜索树的复杂度。在随机数据中，其树高期望为  $\log n$  级别，因此其单次操作的复杂度也是  $O(\log n)$  的。但是出题人不可能次次都出随机数据。因此我们需要使用某种方式来保证其树高不会变得太高。

# 平衡树

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 在之前的操作中，我们并没有关注二叉搜索树的复杂度。在随机数据中，其树高期望为  $\log n$  级别，因此其单次操作的复杂度也是  $O(\log n)$  的。但是出题人不可能次次都出随机数据。因此我们需要使用某种方式来保证其树高不会变得太高。
- 在保证其树高不会变得太高就是让这棵树看上去左右更加地平衡，因此我们称之为平衡树。

# 平衡树

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- Treap 是 tree 和 heap 的合称，其为每个节点生成一个随机的权值，在满足二叉搜索树的性质时，其随机权值需要满足堆的性质。

# 平衡树

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- Treap 是 tree 和 heap 的合称，其为每个节点生成一个随机的权值，在满足二叉搜索树的性质时，其随机权值需要满足堆的性质。
- splay 通过每次访问后将访问到的节点通过某种方式旋转到根节点来保证其复杂度。

# 平衡树

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- Treap 是 tree 和 heap 的合称，其为每个节点生成一个随机的权值，在满足二叉搜索树的性质时，其随机权值需要满足堆的性质。
- splay 通过每次访问后将访问到的节点通过某种方式旋转到根节点来保证其复杂度。
- 替罪羊树通过某种方式来衡量树的不平衡程度，当这棵树过于不平衡时，将其进行拍扁重构。

# 平衡树

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- Treap 是 tree 和 heap 的合称，其为每个节点生成一个随机的权值，在满足二叉搜索树的性质时，其随机权值需要满足堆的性质。
- splay 通过每次访问后将访问到的节点通过某种方式旋转到根节点来保证其复杂度。
- 替罪羊树通过某种方式来衡量树的不平衡程度，当这棵树过于不平衡时，将其进行拍扁重构。
- 由于大部分平衡树的实现难度很大，且常数通常也很大，不建议在 csp、noip 等时间较为紧张的考试中实现平衡树。如果题目允许的话最好用动态开点的权值线段树代替。

# 目录

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

1 倍增

2 求 LCA 的几种算法

3 STL

4 二叉堆

5 二叉搜索树

6 线段树、树状数组



# 树状数组

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- $c_x$  维护的是  $[x - \text{lowbit}(x) + 1, x]$  范围内的和。这样前缀求和以及单点修改只需要访问  $\log n$  个位置。

# 树状数组

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- $c_x$  维护的是  $[x - \text{lowbit}(x) + 1, x]$  范围内的和。这样前缀求和以及单点修改只需要访问  $\log n$  个位置。
- 优点：代码简短，常数小

# 树状数组

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- $c_x$  维护的是  $[x - \text{lowbit}(x) + 1, x]$  范围内的和。这样前缀求和以及单点修改只需要访问  $\log n$  个位置。
- 优点：代码简短，常数小
- 缺点：不容易扩展

# 线段树

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 每个节点表示一个线段，其左右儿子分别代表这个线段的左半部分和右半部分

# 线段树

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 每个节点表示一个线段，其左右儿子分别代表这个线段的左半部分和右半部分
- 进行单点修改、查询或者区间修改、查询只会影响  $\log n$  个区间，其中区间修改需要借助懒标记实现。

# 线段树

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 每个节点表示一个线段，其左右儿子分别代表这个线段的左半部分和右半部分
- 进行单点修改、查询或者区间修改、查询只会影响  $\log n$  个区间，其中区间修改需要借助懒标记实现。
- 代码难度不大（相比平衡树要好写很多），常数也不大

# 线段树

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 每个节点表示一个线段，其左右儿子分别代表这个线段的左半部分和右半部分
- 进行单点修改、查询或者区间修改、查询只会影响  $\log n$  个区间，其中区间修改需要借助懒标记实现。
- 代码难度不大（相比平衡树要好写很多），常数也不大
- 容易扩展

# 树状数组上二分

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 树状数组上二分是一个较为冷门的树状数组应用



# 树状数组上二分

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 树状数组上二分是一个较为冷门的树状数组应用
- 其过程类似于倍增，令  $j$  从  $\log n$  到 0 进行循环，如果可以将当前长度增加  $2^j$  就增加。最终长度就是结果。由于树状数组的性质，判断能否将当前长度增加只需要访问一个树状数组的元素即可。

# NOIP2017 列队

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 题意较为复杂，扔个链接

# NOIP2017 列队

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 题意较为复杂，扔个链接
- <https://www.luogu.com.cn/problem/P3960>

# NOIP2017 列队

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 题意较为复杂，扔个链接
- <https://www.luogu.com.cn/problem/P3960>
- 按照出题人赛后的说法，这道题只有树状数组上二分以及常数很小的线段树上二分能通过，若常数较大或使用平衡树则无法通过。

# NOIP2017 列队

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 首先考虑  $n = 1$  的情况

# NOIP2017 列队

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 首先考虑  $n = 1$  的情况
- 我们可以先在队伍后面把所有询问中出队的人加上，然后令所有位置的权值为 1 表示这个人在队伍中。如果这个人出队了那就把这个权值改为 0

# NOIP2017 列队

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 首先考虑  $n = 1$  的情况
- 我们可以先在队伍后面把所有询问中出队的人加上，然后令所有位置的权值为 1 表示这个人在队伍中。如果这个人出队了那就把这个权值改为 0
- 查找第  $k$  个人就是在树状数组上二分找到第  $k$  个 1，然后将其修改为 0

# NOIP2017 列队

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 对于  $n > 1$  的情况，我们可以对最后一列单独进行  $n = 1$  的算法，求出每次被 **push** 进前  $m - 1$  列的同学，然后再对于每一行进行  $n = 1$  的算法求出离队的同学。



# NOIP2017 列队

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 对于  $n > 1$  的情况，我们可以对最后一列单独进行  $n = 1$  的算法，求出每次被 `push` 进前  $m - 1$  列的同学，然后再对于每一行进行  $n = 1$  的算法求出离队的同学。
- 经过一些精巧的实现，总复杂度  $O(n \log n)$

# JOISC2021 Day1 饮食区

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 题意较为繁琐，丢个链接

# JOISC2021 Day1 饮食区

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 题意较为繁琐，丢个链接
- <https://loj.ac/p/3489>

# JOISC2021 Day1 饮食区

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 按照店铺从小到大的顺序进行遍历，首先不考虑删人的情况，

# JOISC2021 Day1 饮食区

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 按照店铺从小到大的顺序进行遍历，首先不考虑删人的情况，
- 维护一个树状数组表示第  $i$  个时刻（树状数组的下标是时刻）加入的人有多少个加入到当前店铺中，这样区间加人只需要在区间的左端点和右端点  $+1$  处进行单点加减即可。

# JOISC2021 Day1 饮食区

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 按照店铺从小到大的顺序进行遍历，首先不考虑删人的情况，
- 维护一个树状数组表示第  $i$  个时刻（树状数组的下标是时刻）加入的人有多少个加入到当前店铺中，这样区间加人只需要在区间的左端点和右端点  $+1$  处进行单点加减即可。
- 查询时在树状数组上二分即可

# JOISC2021 Day1 饮食区

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 按照店铺从小到大的顺序进行遍历，首先不考虑删人的情况，
- 维护一个树状数组表示第  $i$  个时刻（树状数组的下标是时刻）加入的人有多少个加入到当前店铺中，这样区间加人只需要在区间的左端点和右端点  $+1$  处进行单点加减即可。
- 查询时在树状数组上二分即可
- 接下来考虑加入删人的情况

# JOISC2021 Day1 饮食区

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 我们会发现删人最麻烦的在于删光的情况



# JOISC2021 Day1 饮食区

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 我们会发现删人最麻烦的在于删光的情况
- 因此我们可以假定删光以后继续删，也就是人数变成负数，然后查询的时候把人数加回来即可

# JOISC2021 Day1 饮食区

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 我们会发现删人最麻烦的在于删光的情况
- 因此我们可以假定删光以后继续删，也就是人数变成负数，然后查询的时候把人数加回来即可
- 因此我们需要加上这个位置删人的时候，人数的最小值（可以是负数），然后在查询的时候加回来

# JOISC2021 Day1 饮食区

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 我们会发现删人最麻烦的在于删光的情况
- 因此我们可以假定删光以后继续删，也就是人数变成负数，然后查询的时候把人数加回来即可
- 因此我们需要加上这个位置删人的时候，人数的最小值（可以是负数），然后在查询的时候加回来
- 为了求出人数的最小值，我们还需要维护一个线段树，支持区间加，查询区间  $\min$

# JOISC2021 Day1 饮食区

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 我们会发现删人最麻烦的在于删光的情况
- 因此我们可以假定删光以后继续删，也就是人数变成负数，然后查询的时候把人数加回来即可
- 因此我们需要加上这个位置删人的时候，人数的最小值（可以是负数），然后在查询的时候加回来
- 为了求出人数的最小值，我们还需要维护一个线段树，支持区间加，查询区间  $\min$
- 总时间复杂度  $O(n \log n)$

# 矩形数点问题

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 二维平面上有  $n$  个点，你需要回答  $m$  个询问，每次询问给出一个长宽与坐标轴平行的矩形，然后回答矩形内的点的个数。允许离线。

# 矩形数点问题

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 二维平面上有  $n$  个点，你需要回答  $m$  个询问，每次询问给出一个长宽与坐标轴平行的矩形，然后回答矩形内的点的个数。允许离线。
- $n, m \leq 10^5$

# 矩形数点问题

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 如果点的坐标很大的话可以先将坐标离散化

# 矩形数点问题

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 如果点的坐标很大的话可以先将坐标离散化
- 按照  $y$  坐标从小到大扫描所有点，每次遇到一个点则将其对应的  $x$  坐标处加一。



# 矩形数点问题

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 如果点的坐标很大的话可以先将坐标离散化
- 按照  $y$  坐标从小到大扫描所有点，每次遇到一个点则将其对应的  $x$  坐标处加一。
- 这时我们可以离线求出所有  $lx, rx, 0, ly$  范围内的点的数量。而  $[ly, ry]$  范围内的点就等于  $[0, ry]$  范围内的减去  $[0, ly - 1]$  范围内的点的数量。

# JOISC2021 道路建设

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 给定平面上  $n$  个点对，点的距离是曼哈顿距离。将所有点对距离从小到大排序后输出前  $K$  个。

# JOISC2021 道路建设

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 给定平面上  $n$  个点对，点的距离是曼哈顿距离。将所有点对距离从小到大排序后输出前  $K$  个。
- $n, k \leq 250000$ ，时限 10 秒

# JOISC2021 道路建设

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 首先将坐标轴旋转 45 度，将曼哈顿距离转化为切比雪夫距离

# JOISC2021 道路建设

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 首先将坐标轴旋转 45 度，将曼哈顿距离转化为切比雪夫距离
- 这样满足到一个点距离小于等于  $x$  的点构成一个矩形

# JOISC2021 道路建设

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 首先将坐标轴旋转 45 度，将曼哈顿距离转化为切比雪夫距离
- 这样满足到一个点距离小于等于  $x$  的点构成一个矩形
- 直接按照距离从小到大枚举点对难度较大。因此我们可以考虑换个思路

# JOISC2021 道路建设

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 首先将坐标轴旋转 45 度，将曼哈顿距离转化为切比雪夫距离
- 这样满足到一个点距离小于等于  $x$  的点构成一个矩形
- 直接按照距离从小到大枚举点对难度较大。因此我们可以考虑换个思路
- 考虑二分一个数值，然后求出距离小于等于这个数值的点对数

# JOISC2021 道路建设

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 首先将坐标轴旋转  $45$  度，将曼哈顿距离转化为切比雪夫距离
- 这样满足到一个点距离小于等于  $x$  的点构成一个矩形
- 直接按照距离从小到大枚举点对难度较大。因此我们可以考虑换个思路
- 考虑二分一个数值，然后求出距离小于等于这个数值的点对数
- 这样我们可以唯一确定从小到大第  $K$  个点对的距离，也即距离的上界



# JOISC2021 道路建设

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 首先将坐标轴旋转 45 度，将曼哈顿距离转化为切比雪夫距离
- 这样满足到一个点距离小于等于  $x$  的点构成一个矩形
- 直接按照距离从小到大枚举点对难度较大。因此我们可以考虑换个思路
- 考虑二分一个数值，然后求出距离小于等于这个数值的点对数
- 这样我们可以唯一确定从小到大第  $K$  个点对的距离，也即距离的上界
- 有了确定的距离上界以后，只需要稍微注意一下代码的细节即可枚举出所有距离在范围内的点对。

# JOISC2021 道路建设

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 然后对于确定的距离，求点对数

# JOISC2021 道路建设

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 然后对于确定的距离，求点对数
- 这个问题就是上一道矩形数点问题，可以在  $n \log n$  的时间内解决

# JOISC2021 道路建设

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 然后对于确定的距离，求点对数
- 这个问题就是上一道矩形数点问题，可以在  $n \log n$  的时间内解决
- 总复杂度  $O(n \log n \log X)$

# CF983E NN Country

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 给定一棵  $n$  个点的树，树上有  $m$  条地铁线路（是树上的一条链），你可以花费 1 的代价乘坐任意一条地铁线路到达这条线路上的任意一站。

# CF983E NN Country

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 给定一棵  $n$  个点的树，树上有  $m$  条地铁线路（是树上的一条链），你可以花费 1 的代价乘坐任意一条地铁线路到达这条线路上的任意一站。
- 回答  $q$  次询问，每次询问求出从一个点到达另外一个点需要的最小代价，或输出无法到达。

# CF983E NN Country

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 给定一棵  $n$  个点的树，树上有  $m$  条地铁线路（是树上的一条链），你可以花费 1 的代价乘坐任意一条地铁线路到达这条线路上的任意一站。
- 回答  $q$  次询问，每次询问求出从一个点到达另外一个点需要的最小代价，或输出无法到达。
- $n, m, q \leq 2 * 10^5$

# CF983E NN Country

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 考虑一条链的情况



# CF983E NN Country

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 考虑一条链的情况
- 预处理出每个节点向右乘坐一次地铁最右边能到达哪里

# CF983E NN Country

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 考虑一条链的情况
- 预处理出每个节点向右乘坐一次地铁最右边能到达哪里
- 然后令  $f_{i,j}$  表示从  $i$  开始，乘坐  $2^j$  次地铁，最右边能到达哪里

# CF983E NN Country

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 考虑一条链的情况
- 预处理出每个节点向右乘坐一次地铁最右边能到达哪里
- 然后令  $f_{i,j}$  表示从  $i$  开始, 乘坐  $2^j$  次地铁, 最右边能到达哪里
- 过程和另外几个倍增题差不多

# CF983E NN Country

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 考虑一条链的情况
- 预处理出每个节点向右乘坐一次地铁最右边能到达哪里
- 然后令  $f_{i,j}$  表示从  $i$  开始, 乘坐  $2^j$  次地铁, 最右边能到达哪里
- 过程和另外几个倍增题差不多
- 然后接下来考虑树的情况

# CF983E NN Country

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 我们仍然需要求出每个点往上乘一次地铁最多到达哪里，然后求出倍增数组

# CF983E NN Country

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 我们仍然需要求出每个点往上乘一次地铁最多到达哪里，然后求出倍增数组
- 在进行一次询问时，设起点终点分别为  $u, v$ ，它们的 lca 为  $t$ ，则我们可以借助倍增数组，求出  $u, v$  向上还差一次即可到达  $t$  时的位置  $su, sv$ 。

# CF983E NN Country

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 我们仍然需要求出每个点往上乘一次地铁最多到达哪里，然后求出倍增数组
- 在进行一次询问时，设起点终点分别为  $u, v$ ，它们的 lca 为  $t$ ，则我们可以借助倍增数组，求出  $u, v$  向上还差一次即可到达  $t$  时的位置  $su, sv$ 。
- 于是接下来要么  $su, sv$  全部跑到  $t$ ，需要乘两趟，要么存在一趟地铁，能够使得  $su, sv$  直达。

# CF983E NN Country

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 我们仍然需要求出每个点往上乘一次地铁最多到达哪里，然后求出倍增数组
- 在进行一次询问时，设起点终点分别为  $u, v$ ，它们的 lca 为  $t$ ，则我们可以借助倍增数组，求出  $u, v$  向上还差一次即可到达  $t$  时的位置  $su, sv$ 。
- 于是接下来要么  $su, sv$  全部跑到  $t$ ，需要乘两趟，要么存在一趟地铁，能够使得  $su, sv$  直达。
- 因此问题便转化为了判定是否存在一趟地铁，其路径两端节点分别在  $su, sv$  的子树中



# CF983E NN Country

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 我们仍然需要求出每个点往上乘一次地铁最多到达哪里，然后求出倍增数组
- 在进行一次询问时，设起点终点分别为  $u, v$ ，它们的 lca 为  $t$ ，则我们可以借助倍增数组，求出  $u, v$  向上还差一次即可到达  $t$  时的位置  $su, sv$ 。
- 于是接下来要么  $su, sv$  全部跑到  $t$ ，需要乘两趟，要么存在一趟地铁，能够使得  $su, sv$  直达。
- 因此问题便转化为了判定是否存在一趟地铁，其路径两端节点分别在  $su, sv$  的子树中
- 通过 dfs 序可以将子树的条件转化为一个区间，因此上面的问题转化为了矩形数点问题

# CF983E NN Country

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 我们仍然需要求出每个点往上乘一次地铁最多到达哪里，然后求出倍增数组
- 在进行一次询问时，设起点终点分别为  $u, v$ ，它们的 lca 为  $t$ ，则我们可以借助倍增数组，求出  $u, v$  向上还差一次即可到达  $t$  时的位置  $su, sv$ 。
- 于是接下来要么  $su, sv$  全部跑到  $t$ ，需要乘两趟，要么存在一趟地铁，能够使得  $su, sv$  直达。
- 因此问题便转化为了判定是否存在一趟地铁，其路径两端节点分别在  $su, sv$  的子树中
- 通过 dfs 序可以将子树的条件转化为一个区间，因此上面的问题转化为了矩形数点问题
- 总复杂度  $O(n \log n)$

# CF983E NN Country

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 我们仍然要求出每个点往上乘一次地铁最多到达哪里，然后求出倍增数组

# CF983E NN Country

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 我们仍然需要求出每个点往上乘一次地铁最多到达哪里，然后求出倍增数组
- 在进行一次询问时，设起点终点分别为  $u, v$ ，它们的 lca 为  $t$ ，则我们可以借助倍增数组，求出  $u, v$  向上还差一次即可到达  $t$  时的位置  $su, sv$ 。

# CF983E NN Country

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 我们仍然需要求出每个点往上乘一次地铁最多到达哪里，然后求出倍增数组
- 在进行一次询问时，设起点终点分别为  $u, v$ ，它们的 lca 为  $t$ ，则我们可以借助倍增数组，求出  $u, v$  向上还差一次即可到达  $t$  时的位置  $su, sv$ 。
- 于是接下来要么  $su, sv$  全部跑到  $t$ ，需要乘两趟，要么存在一趟地铁，能够使得  $su, sv$  直达。

# CF983E NN Country

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 我们仍然需要求出每个点往上乘一次地铁最多到达哪里，然后求出倍增数组
- 在进行一次询问时，设起点终点分别为  $u, v$ ，它们的 lca 为  $t$ ，则我们可以借助倍增数组，求出  $u, v$  向上还差一次即可到达  $t$  时的位置  $su, sv$ 。
- 于是接下来要么  $su, sv$  全部跑到  $t$ ，需要乘两趟，要么存在一趟地铁，能够使得  $su, sv$  直达。
- 因此问题便转化为了判定是否存在一趟地铁，其路径两端节点分别在  $su, sv$  的子树中

# CF983E NN Country

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 我们仍然需要求出每个点往上乘一次地铁最多到达哪里，然后求出倍增数组
- 在进行一次询问时，设起点终点分别为  $u, v$ ，它们的 lca 为  $t$ ，则我们可以借助倍增数组，求出  $u, v$  向上还差一次即可到达  $t$  时的位置  $su, sv$ 。
- 于是接下来要么  $su, sv$  全部跑到  $t$ ，需要乘两趟，要么存在一趟地铁，能够使得  $su, sv$  直达。
- 因此问题便转化为了判定是否存在一趟地铁，其路径两端节点分别在  $su, sv$  的子树中
- 通过 dfs 序可以将子树的条件转化为一个区间，因此上面的问题转化为了矩形数点问题

# CF983E NN Country

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 我们仍然需要求出每个点往上乘一次地铁最多到达哪里，然后求出倍增数组
- 在进行一次询问时，设起点终点分别为  $u, v$ ，它们的 lca 为  $t$ ，则我们可以借助倍增数组，求出  $u, v$  向上还差一次即可到达  $t$  时的位置  $su, sv$ 。
- 于是接下来要么  $su, sv$  全部跑到  $t$ ，需要乘两趟，要么存在一趟地铁，能够使得  $su, sv$  直达。
- 因此问题便转化为了判定是否存在一趟地铁，其路径两端节点分别在  $su, sv$  的子树中
- 通过 dfs 序可以将子树的条件转化为一个区间，因此上面的问题转化为了矩形数点问题
- 总复杂度  $O(n \log n)$



# 一道经典题

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 给定一个长度为  $n$  的数组，每次询问可以将一个区间开根号下取整，或者单点修改一个数的值，或者查询区间和

# 一道经典题

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 给定一个长度为  $n$  的数组，每次询问可以将一个区间开根号下取整，或者单点修改一个数的值，或者查询区间和
- $n, q \leq 10^5, 0 \leq a_i \leq 10^{18}$

# 一道经典题

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 一个数字最多被开 6 次根号就会变成 1

# 一道经典题

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 一个数字最多被开 6 次根号就会变成 1
- 因此我们可以在线段树上记录区间最大值，在区间开根号时，如果最大值是 1 就不递归

# 一道经典题

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 一个数字最多被开 6 次根号就会变成 1
- 因此我们可以在线段树上记录区间最大值，在区间开根号时，如果最大值是 1 就不递归
- 这样复杂度是开根号次数乘  $\log n$  的

# CF780G Andryusha and Nervous Barriers

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 在一个  $h * w$  的区域内有  $n$  块横向的板，你会从高处扔下一个球。

# CF780G Andryusha and Nervous Barriers

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 在一个  $h * w$  的区域内有  $n$  块横向的板，你会从高处扔下一个球。
- 球在遇到横向的板时，会裂成两个，分别从这块板的两端落下

# CF780G Andryusha and Nervous Barriers

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 在一个  $h * w$  的区域内有  $n$  块横向的板，你会从高处扔下一个球。
- 球在遇到横向的板时，会裂成两个，分别从这块板的两端落下
- 求落到地面的球的数量，答案对  $10^9 + 7$  取模



# CF780G Andryusha and Nervous Barriers

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 在一个  $h * w$  的区域内有  $n$  块横向的板，你会从高处扔下一个球。
- 球在遇到横向的板时，会裂成两个，分别从这块板的两端落下
- 求落到地面的球的数量，答案对  $10^9 + 7$  取模
- $h \leq 10^9, w \leq 10^5, n \leq 10^5$

# CF780G Andryusha and Nervous Barriers

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 按照从高到低的顺序扫描所有板，过程中使用线段树维护所有横坐标的球的数量

# CF780G Andryusha and Nervous Barriers

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 按照从高到低的顺序扫描所有板，过程中使用线段树维护所有横坐标的球的数量
- 每次遇到一个板时，按照上一题的暴力求出这个区间内所有球的数量并将区间内的球清零

# CF780G Andryusha and Nervous Barriers

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 按照从高到低的顺序扫描所有板，过程中使用线段树维护所有横坐标的球的数量
- 每次遇到一个板时，按照上一题的暴力求出这个区间内所有球的数量并将区间内的球清零
- 然后再在两边的位置加上球的数量即可

# CF780G Andryusha and Nervous Barriers

数据结构相关

戴傅聪

倍增

求 LCA 的几种  
算法

STL

二叉堆

二叉搜索树

线段树、树状  
数组

- 按照从高到低的顺序扫描所有板，过程中使用线段树维护所有横坐标的球的数量
- 每次遇到一个板时，按照上一题的暴力求出这个区间内所有球的数量并将区间内的球清零
- 然后再在两边的位置加上球的数量即可
- 时间复杂度  $O(n \log n)$