分块

faebdc

2018 年 1 月 21 日

分块

分块是什么？

分块

分块是什么？

分块是一种改进的暴力

分块

分块是什么？

分块是一种改进的暴力

比数据结构时间复杂度更差，但比数据结构简单、常数小。

分块

分块是什么？

分块是一种改进的暴力

比数据结构时间复杂度更差，但比数据结构简单、常数小。

甚至很多时候运行起来比数据结构还快。

目录

分块与线段树

单点修改，区间查询

分块与线段树

单点修改，区间查询

区间修改，区间查询（块标记）

块大小的决定

使得复杂度尽可能低

块大小的决定

使得复杂度尽可能低

例如序列总长为n，块大小是t，查询$O(\frac{n\log n}{t})$，修改$O(t)$。

m次询问的总复杂度$O(m(\frac{n\log n}{t}+t))$

块大小的决定

使得复杂度尽可能低

例如序列总长为n，块大小是t，查询$O(\frac{n \log n}{t})$，修改$O(t)$。

m次询问的总复杂度$O(m(\frac{n \log n}{t} + t))$

取$t = \sqrt{n \log n}$，则总复杂度$O(m\sqrt{n \log n})$

块大小的决定

使得复杂度尽可能低

例如序列总长为n，块大小是t，查询$O(\frac{n\log n}{t})$，修改$O(t)$。

m次询问的总复杂度$O(m(\frac{n\log n}{t}+t))$

取$t=\sqrt{n\log n}$，则总复杂度$O(m\sqrt{n\log n})$

很多时候常数的影响难以直接判断，最好的方法就是调整块大小试一试。

bzoj 1012

数列初始为空

两种操作

求末尾L个数中的最大的数，并输出这个数的值。

将n加上t，其中t是最近一次查询操作的答案，并将所得结果对一个固定的常数D取模，将所得答案插入到数列的末尾。

$m \leq 2 \times 10^5$

bzoj 1012

每$\sqrt{m}$个数一个块

插入时判断直接加还是增加新块

bzoj 1012

每$\sqrt{m}$个数一个块

插入时判断直接加还是增加新块

维护每个块的最大值

查询时整块的直接用记录的值，零散的元素暴力。

bzoj 1012

每$\sqrt{m}$个数一个块

插入时判断直接加还是增加新块

维护每个块的最大值

查询时整块的直接用记录的值，零散的元素暴力。

复杂度：$O(m\sqrt{m})$

bzoj 3224

维护一些数，其中需要提供以下操作：

1. 插入x数
2. 删除x数(若有多个相同的数，因只删除一个)
3. 查询x数的排名(若有多个相同的数，因输出最小的排名)
4. 查询排名为x的数
5. 求x的前驱(前驱定义为小于x，且最大的数)
6. 求x的后继(后继定义为大于x，且最小的数)

初始序列为空

$n \leq 10^5$，数的绝对值$\leq 2 \times 10^9$

bzoj 3224

先把数都离散化

bzoj 3224

先把数都离散化

类似权值线段树，记录每个数出现多少次

bzoj 3224

先把数都离散化

类似权值线段树，记录每个数出现多少次

都是线段树基本操作，都可以用分块实现

bzoj 2002

有n个装置，每个装置设定初始弹力系数k_i，当绵羊达到第i个装置时，它会往后弹k_i步，达到第$i+k_i$个装置，若不存在第$i+k_i$个装置，则绵羊被弹飞。

绵羊想知道当它从第i个装置起步时，被弹几次后会被弹飞。（询问操作）

还可以修改某个弹力装置的弹力系数（修改操作）

$n \le 2 \times 10^5, m \le 10^5$

bzoj 2002

将装置分成$\sqrt{n}$块，对每个装置记录需要跳多少次能跳出块，以及跳出块后落在哪个块。

bzoj 2002

将装置分成$\sqrt{n}$块，对每个装置记录需要跳多少次能跳出块，以及跳出块后落在哪个块。

查询：每次都直接到一个新块，总共$O(\sqrt{n})$个块，复杂度$O(\sqrt{n})$。

bzoj 2002

将装置分成$\sqrt{n}$块，对每个装置记录需要跳多少次能跳出块，以及跳出块后落在哪个块。

查询：每次都直接到一个新块，总共$O(\sqrt{n})$个块，复杂度$O(\sqrt{n})$。

修改：只需要修改块内信息，复杂度$O(\sqrt{n})$。

块状链表

块内用数组存

块与块之间用链表形式相连

块状链表

块内用数组存

块与块之间用链表形式相连

若块大小设置为k，

块大小超过$2k$时就分裂成两个块

块状链表

块内用数组存

块与块之间用链表形式相连

若块大小设置为k，

块大小超过$2k$时就分裂成两个块

块大小小于$k/2$时就与相邻块合并

bzoj 3223

维护一个有序数列，其中需要提供以下操作：翻转一个区间，例如原有序序列是5 4 3 2 1，翻转区间是[2,4]的话，结果是5 2 3 4 1。

初始序列有n个数，这个序列依次是$(1, 2, \ldots, n-1, n)$

$n, m \leq 10^5$

bzoj 3223

块状链表维护

bzoj 3223

块状链表维护

翻转一个区间时，零散部分暴力修改，整块的部分直接在链表上进行翻转，并打翻转标记，表示块内元素的正确顺序是反过来。

bzoj 3065

初始一个长度为n的序列，三种操作

插入：插入一个数

修改：修改一个数

查询：查询一个区间中的k小值

强制在线

原序列长度 ≤ 35000

插入个数 ≤ 35000，修改个数 ≤ 70000，查询个数 ≤ 70000 ,$0 \leq$ 每时每刻的权值 ≤ 70000

15s一组数据

bzoj 3065

插入、修改都用块状链表维护

bzoj 3065

插入、修改都用块状链表维护

每一个块都分别在内部用“权值分块”的方法记录每个数出现了多少次（可以认为是分块套分块）。然后“权值分块”的信息要做前缀和。也就是每一个块实际记录的是这个块及以前的块中每个数各出现了多少次

bzoj 3065

插入、修改都用块状链表维护

每一个块都分别在内部用“权值分块”的方法记录每个数出现了多少次（可以认为是分块套分块）。然后“权值分块”的信息要做前缀和。也就是每一个块实际记录的是这个块及以前的块中每个数各出现了多少次

查询时二分，然后利用维护的信息判断二分的大了还是小了。

bzoj 3065

插入、修改都用块状链表维护

每一个块都分别在内部用“权值分块”的方法记录每个数出现了多少次（可以认为是分块套分块）。然后“权值分块”的信息要做前缀和。也就是每一个块实际记录的是这个块及以前的块中每个数各出现了多少次

查询时二分，然后利用维护的信息判断二分的大了还是小了。

每t个数一个块，块内的分块每s个数一个块，修改$O(n/t)$，查询$O(\log n(t+s+n/s))$

bzoj 3065

插入、修改都用块状链表维护

每一个块都分别在内部用“权值分块”的方法记录每个数出现了多少次（可以认为是分块套分块）。然后“权值分块”的信息要做前缀和。也就是每一个块实际记录的是这个块及以前的块中每个数各出现了多少次

查询时二分，然后利用维护的信息判断二分的大了还是小了。

每t个数一个块，块内的分块每s个数一个块，修改$O(n/t)$，查询$O(\log n(t+s+n/s))$

取$s=\sqrt{n}, t=\sqrt{\frac{n}{\log n}}$

bzoj 3065

插入、修改都用块状链表维护

每一个块都分别在内部用“权值分块”的方法记录每个数出现了多少次（可以认为是分块套分块）。然后“权值分块”的信息要做前缀和。也就是每一个块实际记录的是这个块及以前的块中每个数各出现了多少次

查询时二分，然后利用维护的信息判断二分的大了还是小了。

每t个数一个块，块内的分块每s个数一个块，修改$O(n/t)$，查询$O(\log n(t+s+n/s))$

取$s=\sqrt{n}, t=\sqrt{\frac{n}{\log n}}$

平均一次操作复杂度$O(\sqrt{n}\log n)$

bzoj 2038

小Z把这n只袜子从1到N编号

m次操作，每次从编号L到R中随机选择两只。

你的任务便是告诉小Z，他有多大的概率抽到两只颜色相同的袜子。

bzoj 2038

概率就是合法方案数除以总方案数

bzoj 2038

概率就是合法方案数除以总方案数

总方案数很好求，考虑求合法方案数

bzoj 2038 - 莫队

如果现在知道区间$[a, b]$的答案，同时知道区间中每种颜色的袜子有多少只，可以直接计算得到$[a, b+1]$的答案只需考虑$b+1$的情况，如果用线段树维护区间中袜子数量，可以$O(\log n)$把$b+1$的颜色给加上。所以修改是$O(\log n)$的。

bzoj 2038 - 莫队

如果现在知道区间$[a, b]$的答案，同时知道区间中每种颜色的袜子有多少只，可以直接计算得到$[a, b+1]$的答案只需考虑$b+1$的情况，如果用线段树维护区间中袜子数量，可以$O(\log n)$把$b+1$的颜色给加上。所以修改是$O(\log n)$的。

也就是说，可以$O(n \log n)$把区间左或右端点修改一个单位。

bzoj 2038 - 莫队

如果现在知道区间$[a, b]$的答案，同时知道区间中每种颜色的袜子有多少只，可以直接计算得到$[a, b+1]$的答案只需考虑$b+1$的情况，如果用线段树维护区间中袜子数量，可以$O(\log n)$把$b+1$的颜色给加上。所以修改是$O(\log n)$的。

也就是说，可以$O(n \log n)$把区间左或右端点修改一个单位。

将询问排序使得移动次数较少。

bzoj 2038 - 莫队

如果现在知道区间$[a, b]$的答案，同时知道区间中每种颜色的袜子有多少只，可以直接计算得到$[a, b+1]$的答案只需考虑$b+1$的情况，如果用线段树维护区间中袜子数量，可以$O(\log n)$把$b+1$的颜色给加上。所以修改是$O(\log n)$的。

也就是说，可以$O(n\log n)$把区间左或右端点修改一个单位。

将询问排序使得移动次数较少。

$O(\sqrt{n})$分块，左端点所在块为第一关键字，右端点位置为第二关键字，排序，按这个顺序查询。

带修改的莫队

之前的莫队像是二维平面上的移动，因为有修改操作所以把修改的时间也作为一个维度，现在是三维的移动。

带修改的莫队

之前的莫队像是二维平面上的移动，因为有修改操作所以把修改的时间也作为一个维度，现在是三维的移动。

移动左右端点，思路一样。移动时间，就是做出修改。

带修改的莫队

之前的莫队像是二维平面上的移动，因为有修改操作所以把修改的时间也作为一个维度，现在是三维的移动。

移动左右端点，思路一样。移动时间，就是做出修改。

排序：左端点所在块为第一关键字，右端点所在块为第二关键字，时间为第三关键字。

bzoj 2120

一个长为n的序列，两种操作：

询问一个区间里有多少不同的数

修改一个数

$n, m \leq 10000$，修改不超过1000次

bzoj 2120

带修改的莫队来求解

bzoj 1086

国家有n个城市，编号为$1 - n$。一些城市之间有道路相连，任意两个不同的城市之间有且仅有一条直接或间接的道路。为了防止管理太过分散，每个省至少要有B个城市，为了能有效的管理，每个省最多只有3B个城市。每个省必须有一个省会，这个省会可以位于省内，也可以在该省外。但是该省的任意一个城市到达省会所经过的道路上的城市（除了最后一个城市，即该省省会）都必须属于该省。

求一种方案。

$n \leq 1000$

bzoj 1086

DFS，当一个点x的若干子树$\geq B$，就分成一块，x是省会。最后剩下不到B个传上去，用祖先结点划分成块。

bzoj 1086

DFS，当一个点x的若干子树$\geq B$，就分成一块，x是省会。最后剩下不到B个传上去，用祖先结点划分成块。

这样每个块$\leq 2B$

bzoj 1086

DFS，当一个点x的若干子树$\geq B$，就分成一块，x是省会。最后剩下不到B个传上去，用祖先结点划分成块。

这样每个块$\leq 2B$

最后剩下不到B个，

bzoj 1086

DFS，当一个点x的若干子树$\geq B$，就分成一块，x是省会。最后剩下不到B个传上去，用祖先结点划分成块。

这样每个块$\leq 2B$

最后剩下不到B个，放到最后一个块里。

树分块

用上题方法给树分块，类似序列分块的方法维护信息，解决树上问题。

bzoj 3757

树上n个点，每个点有颜色。

有m次询问，每次询问一条路径上有多少种颜色。而且每次询问给出a和b，这个询问认为a和b是同一种颜色。

$n \leq 50000, m \leq 10^5$

bzoj 3757

a, b的限制特判就行。

bzoj 3757

a, b的限制特判就行。

莫队改成树上的，分块方式就是树分块，右端点排序可以按DFS序排。

矩阵分块

分块矩阵的乘法

矩阵分块

分块矩阵的乘法

分块对角矩阵的行列式与逆

分块优化复杂度

± 1 RMQ

分块优化复杂度

± 1 RMQ

Method of Four Russians

Dinic解二分图问题的复杂度分析

增广次数：$O(\sqrt{n})$，对增广路长度分类讨论

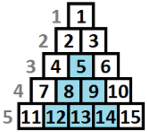
Dinic解二分图问题的复杂度分析

增广次数：$O(\sqrt{n})$，对增广路长度分类讨论

总复杂度：$O(m\sqrt{n})$

Codeforces 354D

有一个n行的金字塔，形如：



可以执行两类操作：

1、花费3的费用对一个格子染色。

2、花费t+2的费用对一个大小为t的子金字塔染色。（例如，上图中蓝色部分就是一个子金字塔。子金字塔必须包含最后一行）。

给出m个格子，要求这m个格子必须被染色，求最小花费。

$n, m \leq 10^5$

Codeforces 354D

m个格子都用第一种方式染色，需要花费$3m$的费用。

Codeforces 354D

m个格子都用第一种方式染色，需要花费3m的费用。

所以对子金字塔染色时，高度最大是$O(\sqrt{n})$级别的，因为太高了花费就会超过3m，一定不是最优的。

Codeforces 354D

m个格子都用第一种方式染色，需要花费3m的费用。

所以对子金字塔染色时，高度最大是$O(\sqrt{n})$级别的，因为太高了花费就会超过3m，一定不是最优的。

所以就可以直接进行DP了。

Codechef COUNTARI

给出一个长度为n的数列A，求有多少个三元组(i, j, k)满足$i < j < k$并且$A[j] - A[i] = A[k] - A[j]$

$n \le 10^5, 1 \le A[i] \le 30000$

Codechef COUNTARI

枚举j，计算多少i, k满足。

Codechef COUNTARI

枚举j，计算多少i,k满足。

这是一个卷积的形式，可以用FFT。

Codechef COUNTARI

枚举j，计算多少i, k满足。

这是一个卷积的形式，可以用FFT。

n遍FFT就更慢了，所以分块，每块t个数

Codechef COUNTARI

枚举j，计算多少i, k满足。

这是一个卷积的形式，可以用FFT。

n遍FFT就更慢了，所以分块，每块t个数

每块只对左右两边做一次FFT，但这样少算了i或k与j在同一个块的情况，这些情况特判就行了。

Codechef COUNTARI

枚举j，计算多少i,k满足。

这是一个卷积的形式，可以用FFT。

n遍FFT就更慢了，所以分块，每块t个数

每块只对左右两边做一次FFT，但这样少算了i或k与j在同一个块的情况，这些情况特判就行了。

取合适的$t=\sqrt{m\log m}$，$O(n\sqrt{m\log m})$

Codechef FNCS

有一个长度为n的数组A，元素编号$1-n$。还有n个函数，函数编号$1-n$，第i个函数的返回值是A中编号在$L[i]$和$R[i]$之间的数的和。

支持两种操作：

将$A[x]$修改为y

求编号在l和r之间的函数的返回值之和

$1 \leq n, m \leq 10^5$

Codechef FNCS

一个函数的返回值可以通过线段树求出。

Codechef FNCS

一个函数的返回值可以通过线段树求出。

对函数按编号进行分块。维护每一块内的函数的返回值之和。

Codechef FNCS

一个函数的返回值可以通过线段树求出。

对函数按编号进行分块。维护每一块内的函数的返回值之和。

预处理出每一块内的函数统计 A 中的每个元素各多少次。

Codechef FNCS

一个函数的返回值可以通过线段树求出。

对函数按编号进行分块。维护每一块内的函数的返回值之和。

预处理出每一块内的函数统计 A 中的每个元素各多少次。

修改 A 中的值时，根据每个块中统计该数的次数，修改每一个块的和。同时修改线段树的值。

Codechef FNCS

一个函数的返回值可以通过线段树求出。

对函数按编号进行分块。维护每一块内的函数的返回值之和。

预处理出每一块内的函数统计 A 中的每个元素各多少次。

修改 A 中的值时，根据每个块中统计该数的次数，修改每一个块的和。同时修改线段树的值。

查询时，整块的部分的和已被计算，其他部分通过线段树逐个函数计算

Codechef GERALD07

有一个无向图G。顶点从1到n标号，边从1到m标号。

有q次询问。对于每次询问，求出当仅保留编号在$[Li, Ri]$中的边时，图G中有多少连通块。

$n, m, q \leq 2 \times 10^5$

Codechef GERALD07

莫队

Codechef GERALD07

莫队

但是，区间缩小怎么办？

Codechef GERALD07

莫队

但是，区间缩小怎么办？

左端点换块就重构并查集。考虑左端点不变的情况。

Codechef GERALD07

莫队

但是，区间缩小怎么办？

左端点换块就重构并查集。考虑左端点不变的情况。

右端点递增，只需考虑左端点的右移。

Codechef GERALD07

莫队

但是，区间缩小怎么办？

左端点换块就重构并查集。考虑左端点不变的情况。

右端点递增，只需考虑左端点的右移。

用类似DFS的方式，在每个询问后左端点还原到块的最右边。

练习题

bzoj4540

bzoj4401

以及任何你想用分块做的数据结构题