分治算法初探

Galaxies

福州第三中学

2017年6月

关于我

- Galaxies（尕拉克西斯）
- Tony Fang
- mailto:Galaxies@vip.qq.com
- http://www.cnblogs.com/galaxies

Tips

- 这些问题可以提问：
  - 听不懂
  - 听不懂
  - 还是听不懂
- 这些问题就别提问了：
  - 卧槽我怎么合卡合爆了！（合x+0比较稳啊）
  - 上银是4+1稳啊还是4+2稳啊？（5+0稳）
  - 我怎么电脑4399关不掉了！（把你电脑关掉呗）
  - 卧槽我开到两个劳尔了啊！（合啊，1+1=0）
- 这些问题自行解决
  - 正常的生理需求
  - 电脑没电了
  - 人没电了

第一部分：基础分治

- bzoj4059 [Cerc2012]Non-boring sequences
- bzoj4025 二分图
- bzoj3237 [AHOI2013]连通图
- 这一部分的内容应该比较基础，是为后面理解分治做准备。请大家保持好连接不要断线。

不无聊的序列

- 给你一个n个数的序列$a_1, a_2, ..., a_n$。定义一个序列是“不无聊的”，当且仅当对于其的每一个子段，都存在一个数在这段内只出现了1次。判断一个数列是不是“不无聊的”。多组数据。
- $1 \leq T \leq 10, 1 \leq n \leq 2 \times 10^5$

不无聊的序列

- 给你一个n个数的序列$a_1, a_2, ..., a_n$。定义一个序列是“不无聊的”，当且仅当对于其的每一个子段，都存在一个数在这段内只出现了1次。判断一个数列是不是“不无聊的”。多组数据。
- $1 \leq T \leq 10, 1 \leq n \leq 2 \times 10^5$
- 考虑分治。定义过程solve(l, r)表示判断区间[l, r]是否满足“不无聊的”，也就是判断包含于区间[l, r]的每个子段是否都有一个数在这段中只出现了一次。我们预处理出每个位置的数前一次出现的位置pre_i和后一次出现的位置nxt_i，然后如果一个位置$k \in [l, r]$满足$pre_i < l, nxt_i > r$，那么任何包含于[l, r]中，经过这个位置的子段都满足条件了。所以就可以分成solve(l, k-1)和solve(k+1, r)来处理了。
- 实现的时候注意找k的时候要从前后一起找，这样类似于“启发式合并”的逆操作，复杂度为$O(nlogn)$。本题总复杂度为$O(Tnlogn)$。

二分图

- 给你一张n个点m条边的图，有T个时刻，第i条边在$(st_i, ed_i]$时刻存在，求每个时刻，这张图是否是二分图。
- $1 \le n \le 10^5, 1 \le m \le 2 \times 10^5, 1 \le T \le 10^5, 1 \le st_i, ed_i \le T$
- Tips: 二分图的判定：没有奇环。可以用并查集实现。

二分图

- 给你一张n个点m条边的图，有T个时刻，第i条边在$(st_i, ed_i]$时刻存在，求每个时刻，这张图是否是二分图。
- $1 \le n \le 10^5, 1 \le m \le 2 \times 10^5, 1 \le T \le 10^5, 1 \le st_i, ed_i \le T$
- Tips: 二分图的判定：没有奇环。可以用并查集实现。
- 对时间分治。定义过程solve(tl, tr, m)表示时刻区间为[tl, tr]，这时刻中出现过的边的集合为e[1..m]。
- 那么对于时刻区间全部出现的边，就直接加入并查集中。然后判定此时是否是二分图，如果已经出现了奇环，显然再怎么加边都不是二分图，所以[tl, tr]的答案为0。否则分成[tl, tmid]和[tmid+1, tr]两个区间分治下去，注意要把当前出现的边的集合分类，可能只出现于第一个区间，也可能只出现于第二个区间，也可能两个都出现了。最后注意要把并查集在这个时刻区间添加的东西撤回（对于其他时刻区间没有用）。

二分图

- 给你一张n个点m条边的图，有T个时刻，第i条边在$(st_i, ed_i]$时刻存在，求每个时刻，这张图是否是二分图。
- $1 \le n \le 10^5, 1 \le m \le 2 \times 10^5, 1 \le T \le 10^5, 1 \le st_i, ed_i \le T$
- 复杂度分析：对于每条边，它们只在$O(logn)$个分治到的时刻区间内出现了（类比于线段树的区间查询），最多在$O(logn)$的时刻区间加入到并查集中，并查集操作需要$O(logn)$的时间，所以总复杂度为$O(mlog^2n)$。

连通图

- 给你一张n个点m条边的图，一共有Q次询问，每次询问删除c_i条边，问整张图是否连通。询问之间独立。
- $1 \le n, Q \le 10^5, 1 \le m \le 2 \times 10^5, 1 \le c_i \le 4$
- 思考：能否类比上题的做法对询问分治？然后把边分类做下去？

连通图

- 给你一张n个点m条边的图，一共有Q次询问，每次询问删除c_i条边，问整张图是否连通。询问之间独立。
- $1 \leq n, Q \leq 10^5, 1 \leq m \leq 2 \times 10^5, 1 \leq c_i \leq 4$
- 思考：能否类比上题的做法对询问分治？然后把边分类做下去？
- 定义过程solve(ql, qr, m)表示询问区间为[ql, qr]，这时刻边的集合为e[1..m]。那么每次我们找出在这个询问区间没有被删除的边，把他添加到并查集中。然后判断这时候是不是连通图，如果已经是，那么[ql, qr]答案为1，即可返回。否则分成[ql, qmid]和[qmid+1, qr]继续分治下去，也要把边的集合分类，把当前添加过的边从集合中删去即可。
- 复杂度分析？

连通图

- 给你一张n个点m条边的图，一共有Q次询问，每次询问删除c_i条边，问整张图是否连通。询问之间独立。
- $1 \leq n, Q \leq 10^5, 1 \leq m \leq 2 \times 10^5, 1 \leq c_i \leq 4$
- 思考：能否类比上题的做法对询问分治？然后把边分类做下去？
- 定义过程solve(ql, qr, m)表示询问区间为[ql, qr]，这时刻边的集合为e[1..m]。那么每次我们找出在这个询问区间没有被删除的边，把他添加到并查集中。然后判断这时候是不是连通图，如果已经是，那么[ql, qr]答案为1，即可返回。否则分成[ql, qmid]和[qmid+1, qr]继续分治下去，也要把边的集合分类，把当前添加过的边从集合中删去即可。
- 复杂度分析？
- 对于每条边，对于每个分治区间，m都有可能达到总边数级别，所以$T(n) = T(n/2) + O((m+n)logn)$，总复杂度为$O(nmlogn)$。
- 这个方法不行！还有其他方法吗？

连通图

- 给你一张n个点m条边的图，一共有Q次询问，每次询问删除c_i条边，问整张图是否连通。询问之间独立。
- $1 \le n, Q \le 10^5, 1 \le m \le 2 \times 10^5, 1 \le c_i \le 4$
- 我们先来寻找为什么会导致复杂度爆炸？原因是：对于每个solve(ql, qr, m)我都要枚举e[1..m]来判断是否添加，m的阶数很可能不变。
- 我们来对应寻找解决办法。

连通图

- 给你一张n个点m条边的图，一共有Q次询问，每次询问删除c_i条边，问整张图是否连通。询问之间独立。
- $1 \leq n, Q \leq 10^5, 1 \leq m \leq 2 \times 10^5, 1 \leq c_i \leq 4$
- 我们先来寻找为什么会导致复杂度爆炸？原因是：对于每个solve(ql, qr, m)我都要枚举e[1..m]来判断是否添加，m的阶数很可能不变。
- 我们来对应寻找解决办法。
- 我们重新定义solve(ql, qr)表示当前处理的询问区间为[ql, qr]。考虑记录cnt[i]表示每条边当前被删去了几次。我们先考虑[qmid+1, qr]所用到的边，很明显[ql, qmid]的边不必删去了，就把对应的cnt更新。每当更新到cnt[i]=0的点，代表这个点在[qmid+1, qr]没有被删除，即可加上这条边，分治solve(qmid+1, qr)，然后撤回操作。接着从[qmid+1, qr]对[ql, qmid]也做一次，分治solve(ql, qmid)，然后撤回操作。
- 复杂度分析：$T(n) = T(n/2) + O(nlogn)$，故总复杂度为$O(nlog^2n)$。

下课时间

- 断线重连时间。
- 提问时间。
- 下面讲的部分，其实和上面的没什么关系（误）。

第二部分：点分治

- poj1741 Trees
- bzoj2152 聪聪可可
- bzoj2599 [IOI2011]Race
- bzoj3697 采药人的路径
- bzoj1758 [WC2010]重建计划

概论

- 处理树上路径问题。每次选取当前连通块的重心，考虑经过重心的路径，然后用重心将连通块分成若干个小连通块，分治下去。
- 最多分$O(logn)$次就只有1个点了。

树

- 求带边权的一棵树中长度不超过K的路径的条数。
- $1 \le n \le 5 \times 10^4, 1 \le w_i, K \le 2 \times 10^5$

树

- 求带边权的一棵树中长度不超过K的路径的条数。
- $1 \le n \le 5 \times 10^4, 1 \le w_i, K \le 2 \times 10^5$
- 考虑对于每个经过重心的路径计算答案。
- 把重心的所有子树的小于等于K的路径全部放入一个数组中，排序后two-pointers扫描数组即可。
- 扣掉路径属于同一个子树（在枚举子树的时候顺便处理即可，方法和上面一模一样）
- 复杂度$O(nlog^2n)$

聪聪可可

- 给一棵带边权的树，求路径长度为3倍数的简单路径出现的概率。
- $1 \leq n \leq 2 \times 10^4, 1 \leq w_i \leq 10^9$

聪聪可可

- 给一棵带边权的树，求路径长度为3倍数的简单路径出现的概率。
- $1 \leq n \leq 2 \times 10^4, 1 \leq w_i \leq 10^9$
- 考虑对于每个经过重心的路径计算答案。
- 把重心的所有子树的路径对于3取模后，全部放入一个数组中，直接组合出3即可（比如余2和余1，余0和余0）。
- 扣掉路径属于同一个子树（在枚举子树的时候顺便处理即可，方法和上面一模一样）。
- 复杂度$O(nlogn)$

赛艇

- 给一棵带边权的树，求一个边最少的简单路径，权值为K。
- $1 \leq n \leq 2 \times 10^5, 0 \leq K \leq 10^6$

赛艇

- 给一棵带边权的树，求一个边最少的简单路径，权值为K。
- $1 \leq n \leq 2 \times 10^5, 0 \leq K \leq 10^6$
- 套用刚刚的做法？
- 随机一位同学上来讲下。
- 复杂度分析？

赛艇

- 给一棵带边权的树，求一个边最少的简单路径，权值为K。
- $1 \leq n \leq 2 \times 10^5, 0 \leq K \leq 10^6$
- 套用刚刚的做法？
- 随机一位同学上来讲下。
- 复杂度分析？
- 应该使用子树合并来完成本题
- 使用子树合并的时候注意要按照子树深度来依次合并！
- 这样就不用扣除重复情况了，是点分治的一般思路。清空不要用memset！
- 复杂度$O(nlog^2n)$。

采药人的路径

- 采药人的药田是一个树状结构，每条路径上都种植着同种药材。
- 采药人以自己对药材独到的见解，对每种药材进行了分类。大致分为两类，一种是阴性的，一种是阳性的。
- 采药人每天都要进行采药活动。他选择的路径是很有讲究的，他认为阴阳平衡是很重要的，所以他走的一定是两种药材数目相等的路径。采药工作是很辛苦的，所以他希望他选出的路径中有一个可以作为休息站的节点（不包括起点和终点），满足起点到休息站和休息站到终点的路径也是阴阳平衡的。他想知道他一共可以选择多少种不同的路径。
- $1 \le n \le 10^5$

采药人的路径

- 采药人的药田是一个树状结构，每条路径上都种植着同种药材。
- 采药人以自己对药材独到的见解，对每种药材进行了分类。大致分为两类，一种是阴性的，一种是阳性的。
- 采药人每天都要进行采药活动。他选择的路径是很有讲究的，他认为阴阳平衡是很重要的，所以他走的一定是两种药材数目相等的路径。采药工作是很辛苦的，所以他希望他选出的路径中有一个可以作为休息站的节点（不包括起点和终点），满足起点到休息站和休息站到终点的路径也是阴阳平衡的。他想知道他一共可以选择多少种不同的路径。
- $1 \leq n \leq 10^5$
- 两种药材看成1和-1，f[i][0/1]表示前面几个子树前缀和为i，是否存在前缀和为0的点。处理出子树的这个值，然后子树合并即可。其中i的范围是[-n,n]，实现需要注意细节。
- 注意子树合并按照深度顺序（从浅到深）!
- 复杂度$O(nlogn)$。

重建计划

- 给一棵带有边权的树，求一个经过边数量在[L,U]，且满足路径上的平均权值最大的路径。L和U的设定将保证有解。
- $1 \le n \le 10^5, 0 < w_i \le 10^6$

重建计划

- 给一棵带有边权的树，求一个经过边数量在[L,U]，且满足路径上的平均权值最大的路径。L和U的设定将保证有解。
- $1 \leq n \leq 10^5, 0 < w_i \leq 10^6$
- 二分答案，将权值改成原来权值减去二分的值，只要判断是否存在经过边数[L,U]的路径权值和即可。
- 点分治子树合并，f_i记录路径长度为i的最大权值，按照深度顺序（从浅到深）进行操作。每次相当于扫描f数组和当前子树的g数组，是否存在$f_x + g_y \geq 0$，且$L \leq x + y \leq U$使用单调队列即可将单次复杂度降为$O(MaxDepth)$，这样点分治总复杂度就是$O(nlogn)$。
- 加上二分的复杂度为$O(nlog^2n)$
- 但是……过不去！优化？

重建计划

- 给一棵带有边权的树，求一个经过边数量在[L,U]，且满足路径上的平均权值最大的路径。L和U的设定将保证有解。
- $1 \le n \le 10^5, 0 < w_i \le 10^6$
- 考虑点分治部分有没有重复做很多次的事情？

重建计划

- 给一棵带有边权的树，求一个经过边数量在[L,U]，且满足路径上的平均权值最大的路径。L和U的设定将保证有解。
- $1 \leq n \leq 10^5, 0 < w_i \leq 10^6$
- 考虑点分治部分有没有重复做很多次的事情？
- 对！找重心，我们先预处理“做”一次点分治，把重心顺序记录下来，接下来我们只要用一个for循环来模拟每次找重心的点分治即可。
- 时间复杂度不变，仍为$O(nlog^2n)$，但是加了很大的优化，加上读入优化总时间只有27s（总时限40s）。

下课时间

- 断线重连时间。
- 提问时间。
- 下面讲的部分，其实和上面的没什么关系（误），但是和第一部分有关系。

第三部分：整体二分和CDQ分治

- bzoj2527 [Poi2011]Meteors
- bzoj3110 [Zjoi2013]K大数查询
- bzoj2738 矩阵乘法
- bzoj3262 陌上花开
- bzoj4430 [Nwerc2015]Guessing Camels
- bzoj3939 [Usaco2015 Feb]Cow Hopscotch

概论

- 整体二分就是：你发现一个东西可以二分答案，但是题目问你一坨东西，有些时候你就可以把他们放在一起二分，叫做整体二分。
- CDQ分治就是：把操作序列分成两半，分治下去并处理前一半对后一半的影响。
- 经常套用的数据结构是BIT（常数小）。

米提尔斯

- 某组织有N个成员国。现在它发现了一颗新的星球，这颗星球的轨道被分为M份（第M份和第1份相邻），第i份上有第A_i个国家的太空站。这个星球经常会下陨石雨。该组织已经预测了接下来K场陨石雨的情况。该组织的第i个成员国希望能够收集P_i单位的陨石样本。你的任务是判断对于每个国家，它需要在第几次陨石雨之后，才能收集足够的陨石。
- $1 \le N, M, K \le 3 \times 10^5, 1 \le P_i \le 10^9$

米提尔斯

- 某组织有N个成员国。现在它发现了一颗新的星球，这颗星球的轨道被分为M份（第M份和第1份相邻），第i份上有第A_i个国家的太空站。这个星球经常会下陨石雨。该组织已经预测了接下来K场陨石雨的情况。该组织的第i个成员国希望能够收集P_i单位的陨石样本。你的任务是判断对于每个国家，它需要在第几次陨石雨之后，才能收集足够的陨石。
- $1 \le N, M, K \le 3 \times 10^5, 1 \le P_i \le 10^9$
- 我们对于一个国家明显可以二分答案然后判断呀。
- 那么对于一坨国家……也可以啊!

米提尔斯

- 某组织有N个成员国。现在它发现了一颗新的星球，这颗星球的轨道被分为M份（第M份和第1份相邻），第i份上有第A_i个国家的太空站。这个星球经常会下陨石雨。该组织已经预测了接下来K场陨石雨的情况。该组织的第i个成员国希望能够收集P_i单位的陨石样本。你的任务是判断对于每个国家，它需要在第几次陨石雨之后，才能收集足够的陨石。
- $1 \le N, M, K \le 3 \times 10^5, 1 \le P_i \le 10^9$
- 我们对于一个国家明显可以二分答案然后判断呀。
- 那么对于一坨国家……也可以啊！
- 定义solve(l, r, al, ar)表示二分到[l, r]，当前答案在这个区间的国家是[al, ar]。我们每次模拟[l, mid]的操作，然后看看哪些国家达到目标/还没达到，划分到两边，设分界点为pos。那么进行分治：solve(l, mid, al, pos)和solve(mid+1, r, pos+1, ar)。注意本题统计的时候每个国家的太空站用边链表连起来，访问方便。注意要撤销操作并且记录当前值。
- 操作相当于区间加，单点查询，用BIT即可。复杂度$O(nlog^2n)$。

K大数查询

- 有N个位置，M个操作。操作有两种，每次操作要么是在第a个位置到第b个位置，每个位置加入一个数c；要么是询问从第a个位置到第b个位置，第k大的数是多少。
- $1 \le N, M \le 5 \times 10^4, 1 \le |c| \le N, 0 \le k \le 2^{31} - 1$

K大数查询

- 有N个位置，M个操作。操作有两种，每次操作要么是在第a个位置到第b个位置，每个位置加入一个数c；要么是询问从第a个位置到第b个位置，第k大的数是多少。
- $1 \le N, M \le 5 \times 10^4, 1 \le |c| \le N, 0 \le k \le 2^{31} - 1$
- 考虑二分查找第K大数的方法，把它应用到整体二分即可。
- 定义solve(l, r, al, ar)表示二分到[l, r]，当前答案在这个区间的询问是[al, ar]。我们每次模拟[l, mid]的操作，然后看看满足了哪些询问，划分到两边，设分界点为pos。那么进行分治：solve(l, mid, al, pos)和solve(mid+1, r, pos+1, ar)。注意划分的时候不能影响相对顺序。注意要撤销操作并且记录当前值。
- 操作相当于区间加，区间查询，用两棵BIT即可实现。
- 时间复杂度$O(nlog^2n)$。

矩阵乘法

- 给你一个$n \times n$的矩阵，不用算矩阵乘法，但是每次询问一个子矩形的第K小数。
- $1 \le n \le 750, 1 \le K \le n^2, 0 \le a_{i,j} \le 10^9$

矩阵乘法

- 给你一个$n \times n$的矩阵，不用算矩阵乘法，但是每次询问一个子矩形的第K小数。
- $1 \leq n \leq 750, 1 \leq K \leq n^2, 0 \leq a_{i,j} \leq 10^9$
- 考虑二分查找第K小数的方法，把它应用到整体二分即可。按照权值顺序往矩阵里一步步加数。
- 定义solve(l, r, al, ar, fr)表示二分到[l, r]，当前答案在这个区间的询问是[al, ar]，数加到fr了。我们每次加数一直加到mid，然后看看满足了哪些询问，划分到两边，设分界点为pos。那么进行分治：solve(l, mid, al, pos)和solve(mid+1, r, pos+1, ar)。注意要撤销操作并且记录当前值。
- 操作相当于二维单点加，区间查询，用二维BIT即可实现。
- 时间复杂度$O(n^2log^3n)$。反正能过。

陌上花开

- 有n朵花，每朵花有三个属性：花形(s)、颜色(c)、气味(m)，又三个整数表示。现要对每朵花评级，一朵花的级别是它拥有的美丽能超过的花的数量。定义一朵花A比另一朵花B要美丽，当且仅当$s_a \geq s_b, c_a \geq c_b, m_a \geq m_b$。显然，两朵花可能有同样的属性。需要统计出评出每个等级的花的数量。
- $1 \leq n \leq 10^5, 1 \leq K \leq 2 \times 10^5, 0 \leq s_i, c_i, m_i \leq K$

陌上花开

- 有n朵花，每朵花有三个属性：花形(s)、颜色(c)、气味(m)，又三个整数表示。现要对每朵花评级，一朵花的级别是它拥有的美丽能超过的花的数量。定义一朵花A比另一朵花B要美丽，当且仅当$s_a \geq s_b, c_a \geq c_b, m_a \geq m_b$。显然，两朵花可能有同样的属性。需要统计出评出每个等级的花的数量。
- $1 \leq n \leq 10^5, 1 \leq K \leq 2 \times 10^5, 0 \leq s_i, c_i, m_i \leq K$
- 这个问题叫三维偏序。给出三维空间内的若干个点，求一个长方体空间内的点的个数。一维排序，一维CDQ分治，一维BIT即可解决。时间复杂度$O(nlog^2n)$。
- 先整合属性相等的花，然后按c值排序。定义solve(l, r)表示现在处理的是s值在[l,r]中的花，分治的时候把s值属于[l,mid]的合s值属于[mid+1,r]的划分，然后接在一起，保持相对顺序。把s值在[l,mid]的花的影响放入s值在[mid+1,r]的花的等级中，这一步由于排序后没有改变相对顺序，所以可以用two-pointers来扫描花。然后把花的m值放入BIT中，每次就是统计BIT中小于等于当前m的花的个数，记得撤回BIT的操作。然后知道等级就很好算答案了。

盖斯因·坎梅尔斯

- 给你三个人赌骆驼赛跑的名单，求有多少对骆驼在每个人的名单中顺序相同。
- $1 \leq n \leq 2 \times 10^5, 1 \leq a_i, b_i, c_i \leq n$

盖斯因·坎梅尔斯

- 给你三个人赌骆驼赛跑的名单，求有多少对骆驼在每个人的名单中顺序相同。
- $1 \leq n \leq 2 \times 10^5, 1 \leq a_i, b_i, c_i \leq n$
- 可以转化为三维偏序问题，用在a,b,c中的位置作为三维坐标。一维排序，一维CDQ分治，一维BIT即可解决。时间复杂度$O(nlog^2n)$。

靠·霍珀斯科瞿

- 游戏在一个$R \times C$的网格上进行，每个格子有一个取值在[1,k]之间的整数标号，奶牛开始在左上角的格子，目的是通过若干次跳跃后到达右下角的格子，当且仅当格子A和格子B满足如下条件时能从格子A跳到格子B：
  - B格子在A格子的严格右方(B的列号严格大于A的列号)
  - B格子在A格子的严格下方(B的行号严格大于A的行号)
  - B格子的标号和A格子的标号不同
- 请你帮助奶牛计算出从左上角的格子到右下角的格子一共有多少种不同的方案
- $1 \leq R, C \leq 750, 1 \leq k \leq R \times C$

靠·霍珀斯科瞿

- 游戏在一个$R \times C$的网格上进行，每个格子有一个取值在[1,k]之间的整数标号，奶牛开始在左上角的格子，目的是通过若干次跳跃后到达右下角的格子，当且仅当格子A和格子B满足如下条件时能从格子A跳到格子B：
  - B格子在A格子的严格右方(B的列号严格大于A的列号)
  - B格子在A格子的严格下方(B的行号严格大于A的行号)
  - B格子的标号和A格子的标号不同
- 请你帮助奶牛计算出从左上角的格子到右下角的格子一共有多少种不同的方案
- $1 \le R, C \le 750, 1 \le k \le R \times C$
- CDQ分治x坐标，然后先分治到y坐标处理出dp值，再考虑x坐标在上方的对于x坐标在下方的贡献。对于y坐标扫描一遍，记录标号为i的dp值之和s[i]，以及之前的所有dp值之和all，那么这个地方的dp值就是all - s[a[i,j]]。然后再分治右边部分。
- 时间复杂度$O(RClog(RC))$。

总结

- 分治算法是OI中一种经典算法。需要理解并灵活加以运用。
- 以上代码希望回去自己实现一遍。
- 实现有困难可以看bzoj的提交记录中的AC代码。或参照我的博客也有AC代码。

致谢

- 谢谢大家的聆听!