

算法优化策略

-----anantheparty



算法优化策略

- 差分，前缀和
 - 多维差分前缀和
 - 树上差分前缀和
- 离散化
- 扫描线
- 2-pointer
- 悬线法
 - 单调栈
- 二分&贪心
- 其他优化策略



复杂度

- 时间复杂度和空间复杂度是衡量一个算法效率的重要标准
- 时间复杂度：该算法用时（基本操作次数）随数据规模而增长的趋势
 - 通常用大O表示时间复杂度上届
 - 实际应用中，还需要考虑O相等时的常数影响。
- 空间复杂度：该算法用空间随数据规模而增长的趋势



前缀和

- 前缀和：简单可以理解为一段数列的前x项和
 - 扩展到二维理解为：某点左上角区域所有数字和
 - 扩展到树上理解为：某点到根节点的数字和
- 前缀和数组：B[i]表示A数组第1项到第i项的和

$$B[i] = B[i-1] + A[i]$$



前缀和

- 给你一个序列， m 次询问，每次询问 $[l,r]$ 的和/积



前缀和

- 给你一个二维平面， m 次询问，每次求一个矩形的数字和



前缀和

- 给你一个树，m次询问，权值在点上，每次求一条链上的数字和
- 给你一个树，m次询问，权值在链上，每次求一条链上的数字和



前缀和

- 给你一个序列， m 次询问，每次询问 $[1, l]$ 或 $[r, n]$ 的最大值



差分

- 差分：求两项之间的差，为前缀和的逆运算

$$B[i] = B[i-1] + A[i]$$

$$A[i] = B[i] - B[i-1]$$

- 树差分
 - 点差分：差分值存在点上，表示该点与父亲的差
 - 边差分：差分值存在边上，表示该边父节点和子节点的差



差分

- 给你一个序列， m 次修改，每次对 $[l,r]$ 执行区间加，最后询问所得序列



差分

- 在 $n \times n$ 的格子中有 m 个矩形地毯。
- 给出这些地毯的信息，问每个点被多少个地毯覆盖。
- 每个地毯给出左上角和右下角坐标



差分

- 给你一棵树，每次有一些链的区间加操作，求最后每个点的权值



2012D2T2 P1083 借教室

- [Link](#)

题目描述

[展开](#)

在大学期间，经常需要租借教室。大到院系举办活动，小到学习小组自习讨论，都需要向学校申请借教室。教室的大小功能不同，借教室人的身份不同，借教室的手续也不一样。

面对海量租借教室的信息，我们自然希望编程解决这个问题。

我们需要处理接下来 n 天的借教室信息，其中第 i 天学校有 r_i 个教室可供租借。共有 m 份订单，每份订单用三个正整数描述，分别为 d_j, s_j, t_j ，表示某租借者需要从第 s_j 天到第 t_j 天租借教室（包括第 s_j 天和第 t_j 天），每天需要租借 d_j 个教室。

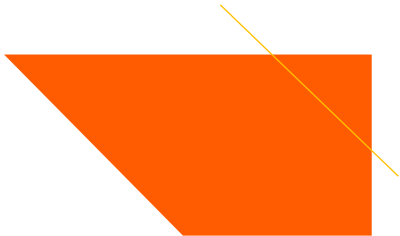
我们假定，租借者对教室的大小、地点没有要求。即对于每份订单，我们只需要每天提供 d_j 个教室，而它们具体是哪些教室，每天是否是相同的教室则不用考虑。

借教室的原则是先到先得，也就是说我们要按照订单的先后顺序依次为每份订单分配教室。如果在分配的过程中遇到一份订单无法完全满足，则需要停止教室的分配，通知当前申请人修改订单。这里的无法满足指从第 s_j 天到第 t_j 天中有至少一天剩余的教室数量不足 d_j 个。

现在我们需要知道，是否会有订单无法完全满足。如果有，需要通知哪一个申请人修改订单。



2012D2T2 P1083 借教室 eazy+



P3664

● [Link](#)

输入 #1

复制

```
4
2 2 3 0
2 7 3 7
2 7 7 7
0 0 0 0
```

输出 #1

复制

14

小TY突然想画画，他有独特的艺术风格，他从 $N \times N$ 空白画布开始，其中0表示画布的空单元格。然后他会在画布上绘制恰好矩形，每个颜色是1到 $N \times N$ 中的一个。他每次可以选择任意一种未使用过的颜色进行绘画。例如， he 可以从颜色2的矩形开始，画出这样的画布：

2 2 2 0

2 2 2 0

2 2 2 0

0 0 0 0

然后 he 可以用颜色7绘制一个矩形：

2 2 2 0

2 7 7 7

2 7 7 7

0 0 0 0

然后 he 可以在颜色3上绘制一个小矩形：

2 2 3 0

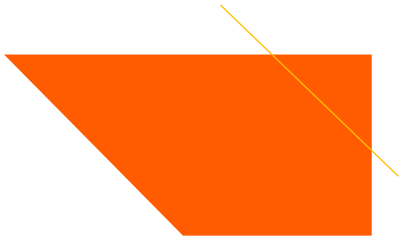
2 7 3 7

2 7 7 7

0 0 0 0

每个矩形都平行于画布边缘，而且矩形可以与整个画布一样大或者像一个单元一样小。每个颜色从1到正好使用一次，后来的颜色可能完全覆盖一些较早画上的颜色。

现在已知画布的最终状态，请计算有多少种颜色可能被第一个被画。



P3664



离散化

- 离散化本质上可以看成是一种哈希，其保证数据在哈希以后仍然保持原来的全/偏序关系。
- 当有些数据因为本身很大或者类型不支持，自身无法作为数组的下标来方便地处理，而影响最终结果的只有元素之间的相对大小关系时，我们可以将原来的数据按照从大到小编号来处理问题，即离散化。
- 常见处理：大整数、浮点数、字符串
- 偷懒处理：使用map



离散化

- OI中常见具体应用：
 - 需要按权值为数组下标进行某种操作，但权值较大
 - 偷懒操作：map，每次操作复杂度 $O(\log n)$
 - 离散化：初始化小常数 $n \log$ ，每次操作 $O(1)$
 - 线段树区间较大



离散化

- STL简易完成离散化操作

```
std::sort(a + 1, a + 1 + n);
```

```
len = std::unique(a + 1, a + n + 1) - a - 1;
```

```
for (int i = 0; i < n; ++i)
```

```
    b[i] = std::lower_bound(a + 1, a + len + 1, b[i]) - a;
```



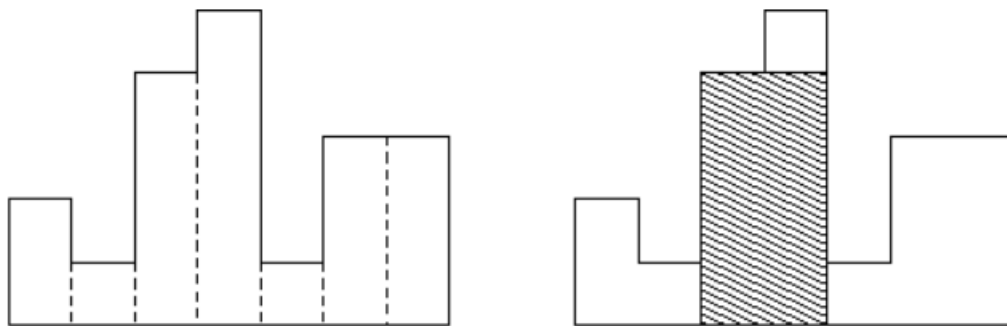
离散化

- 手写离散化

```
sort(b+1,b+n+1);  
for(int i=1;i<=n;i++)  
{  
    if(b[i+1]==b[i])continue;  
    c[++cnt]=b[i];  
}  
for(int i=1;i<=n;i++)  
    d[i]=(lower_bound(c+1,c+cnt+1,a[i])-c);
```

悬线法

- SP1805



如图所示，在一条水平线上有 n 个宽为 1 的矩形，求包含于这些矩形的最大子矩形面积（图中的阴影部分的面积即所求答案）。



单调栈

- 悬线法能解决的问题单调栈都能解决，但悬线法更容易理解

P4147

● [Link](#)

题目背景

[展开](#)

有一天，小猫 rainbow 和 freda 来到了湘西张家界的天门山玉蟾宫，玉蟾宫宫主蓝兔盛情地款待了它们，并赐予它们一片土地。

题目描述

这片土地被分成 $N \times M$ 个格子，每个格子里写着 'R' 或者 'F'，R 代表这块土地被赐予了 rainbow，F 代表这块土地被赐予了 freda。

现在 freda 要在这里卖萌。。。它要找一块矩形土地，要求这片土地都标着 'F' 并且面积最大。

但是 rainbow 和 freda 的 OI 水平都弱爆了，找不出这块土地，而蓝兔也想看 freda 卖萌（她显然是不会编程的.....），所以它们决定，如果你找到的土地面积为 S ，它们每人给你 S 两银子。

输入输出样例

输入 #1

[复制](#)

```
5 6
R F F F F F
F F F F F F
R R R F F F
F F F F F F
F F F F F F
```

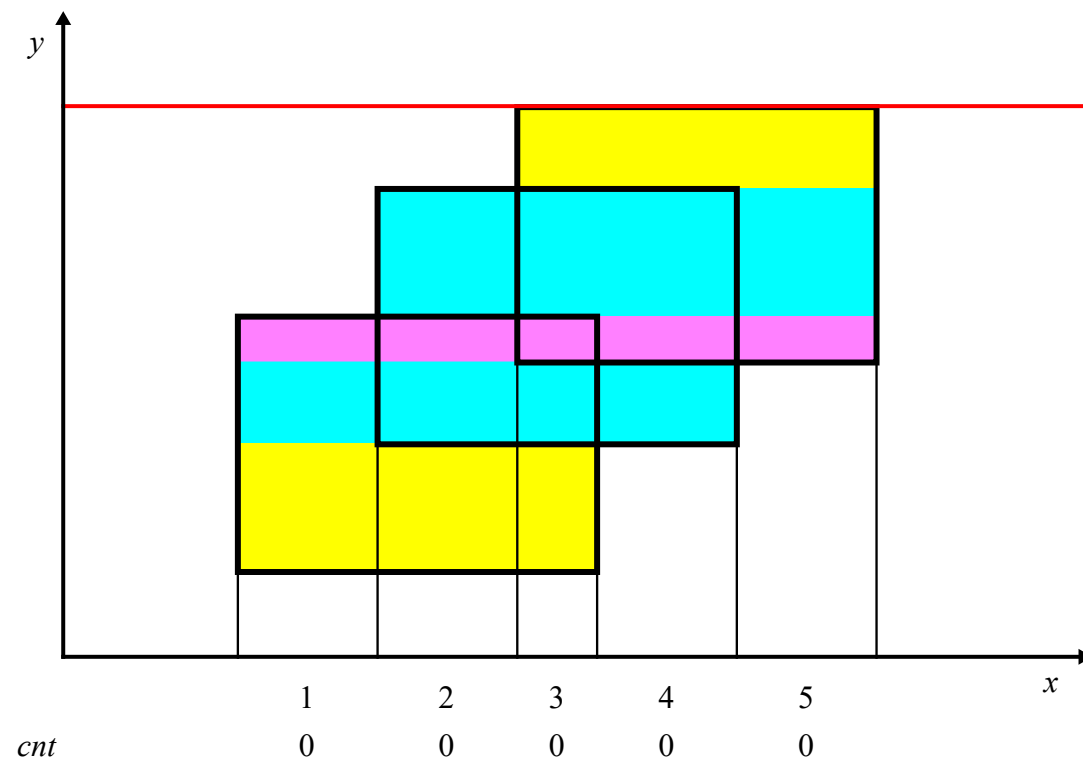
输出 #1

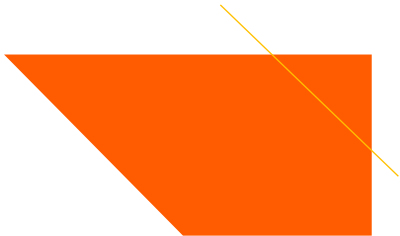
[复制](#)

```
45
```

扫描线

- 本质是一种二维差分
- 通常需要离散化





P3664

● [Link](#)

输入 #1

复制

```
4
2 2 3 0
2 7 3 7
2 7 7 7
0 0 0 0
```

输出 #1

复制

14

小TY突然想画画，他有独特的艺术风格，他从 $N \times N$ 空白画布开始，其中0表示画布的空单元格。然后他会在画布上绘制恰好矩形，每个颜色是1到 $N \times N$ 中的一个。他每次可以选择任意一种未使用过的颜色进行绘画。例如， he 可以从颜色2的矩形开始，画出这样的画布：

2 2 2 0

2 2 2 0

2 2 2 0

0 0 0 0

然后 he 可以用颜色7绘制一个矩形：

2 2 2 0

2 7 7 7

2 7 7 7

0 0 0 0

然后 he 可以在颜色3上绘制一个小矩形：

2 2 3 0

2 7 3 7

2 7 7 7

0 0 0 0

每个矩形都平行于画布边缘，而且矩形可以与整个画布一样大或者像一个单元一样小。每个颜色从1到正好使用一次，后来的颜色可能完全覆盖一些较早画上的颜色。

现在已知画布的最终状态，请计算有多少种颜色可能被第一个被画。



P3664



2-pointer

- Two sum:给你一个数列，找到两个数，使它们和为x



2-pointer

- Three sum:给你一个数列, 找到三个数, 使它们和为x



UVA11292

题意翻译

- [Link](#)

你的王国里有一条 n 个头的恶龙，你希望雇佣一些骑士把它杀死（即砍掉所有头）。村里有 m 个骑士可以雇佣，一个能力值为 x 的骑士可以砍掉恶龙一个直径不超过 x 的头，且需要支付 x 个金币。如何雇佣骑士才能砍掉龙的所有头，且需要支付的金币最少？注意，一个骑士只能砍一个头。（且不能被雇佣两次）。输入格式

输入包含多组数据。每组数据的第一行为正整数 n 和 m （ $1 \leq n, m \leq 20000$ ）；以下 n 行每行为一个整数，即恶龙每个头的直径；以下 m 行每行为一个整数，即每个骑士的能力。输入结束标志为 $n=m=0$ 。输出格式

对于每组数据，输出最小花费。如果无解，输出“Loowater is doomed!”。

感谢@ACdreamer 提供的翻译



2-pointer 滑动窗口

- 给定一个长度为 n 的数列和一个正整数 s , 请找出该数列中和 $\geq s$ 的最小子区间。



2-pointer

- 当这两个指针作为一个区间的左右节点时，我们称这种思想为：滑动窗口
- 通常解决的问题：有一定约束条件下选择最优区间的情况



P1638 easy

- 长度为 n 的序列, m 种数
- 找一个最短区间, 使得所有数出现一遍
- $n \leq 1e6, m \leq 2e3$



P5745

- [Link](#)



P3143

- [Link](#)

奶牛Bessie很喜欢闪亮亮的东西（Baling~Baling~），所以她喜欢在她的空余时间开采钻石！她现在已经收集了N颗不同大小的钻石（ $N \leq 50,000$ ），现在她想在谷仓的两个陈列架上摆放一些钻石。

Bessie想让这些陈列架上的钻石保持相似的大小，所以她不会把两个大小相差K以上的钻石同时放在一个陈列架上（如果两颗钻石的大小差值为K，那么它们可以同时放在一个陈列架上）。现在给出K，请你帮Bessie确定她最多一共可以放多少颗钻石在这两个陈列架上。

输入输出样例

输入 #1

复制

```
7 3
10
5
1
12
9
5
14
```

输出 #1

复制

```
5
```



二分&贪心

- 二分：目标是否具有二分性
- 贪心：该贪心是否正确？



其他优化策略

- 离线/在线算法（提高组不常见）：
 - 在线：每次询问后，立马可以得到查询结果
 - 离线：需要知道所有要查询的值，然后一口气查询出所有结果

如果题目询问的内容有多个，一起回答比一个一个回答轻松不少，可以试试先将问题存起来，一口气回答

多见于将提问的某一维排序后，按照顺序解决比较轻松
(不要忘记再将顺序排回去)



其他优化策略

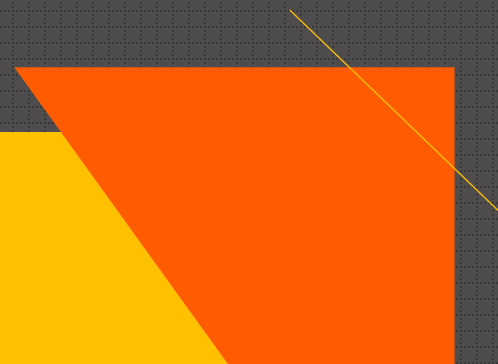
- 随机化：部分结构在随机化下会有期望不错的结果，如随机顺序插入的二叉搜索树深度为 \log 级别，随机后的快速排序期望复杂度为 $n\log n$ ，当顺序不影响题目时，为何不给自己的暴力添加一个随机化呢？

`std::shuffle(first, last, myrand)` (C++11)

- 随机化算法：模拟退火

题目要求最优解，但你仅可以算出一个状态的得分（比较两个解的优劣），不能求到全局最优解，用模拟退火求得的较优解，大概率是最优

杂题选讲





P6602 P3029

- [Link](#)



P1360

[Link](#)



P3067

- 给 n 个数，从中任意选出一些数，使这些数能分成和相等的两组。
求有多少种选数的方案。

$2 \leq N \leq 30$

2016D1T2 P1600 天天爱跑步

[Link](#)

题目描述

[展开](#)

小c 同学认为跑步非常有趣，于是决定制作一款叫做《天天爱跑步》的游戏。《天天爱跑步》是一个养成类游戏，需要玩家每天按时上线，完成打卡任务。

这个游戏的地图可以看作——一棵包含 n 个结点和 $n - 1$ 条边的树，每条边连接两个结点，且任意两个结点存在一条路径互相可达。树上结点编号为从 1 到 n 的连续正整数。

现在有 m 个玩家，第 i 个玩家的起点为 s_i ，终点为 t_i 。每天打卡任务开始时，所有玩家在第 0 秒同时从自己的起点出发，以每秒跑一条边的速度，不间断地沿着最短路径向着自己的终点跑去，跑到终点后该玩家就算完成了打卡任务。（由于地图是一棵树，所以每个人的路径是唯一的）

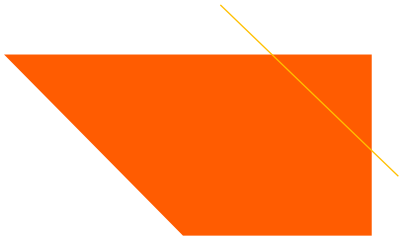
小c 想知道游戏的活跃度，所以在每个结点上都放置了一个观察员。在结点 j 的观察员会选择在第 w_j 秒观察玩家，一个玩家能被这个观察员观察到当且仅当该玩家在第 w_j 秒也正好到达了结点 j 。小c 想知道每个观察员会观察到多少人？

注意：我们认为一个玩家到达自己的终点后该玩家就会结束游戏，他不能等待一段时间后再被观察员观察到。即对于把结点 j 作为终点的玩家：若他在第 w_j 秒前到达终点，则在结点 j 的观察员不能观察到该玩家；若他正好在第 w_j 秒到达终点，则在结点 j 的观察员可以观察到这个玩家。



2016D1T2 P1600 天天爱跑步

[Link](#)

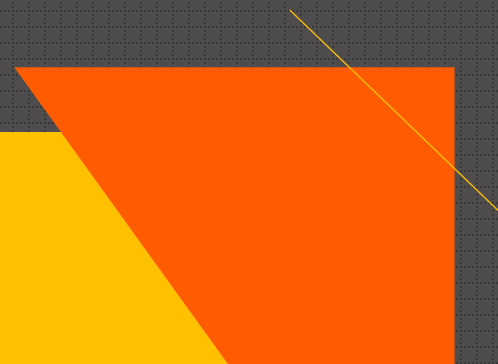


2016D1T2 P1600 天天爱跑步

$O(N \log n)$ \Rightarrow LCA

2016D1T2 P1600 天天爱跑步

作业



2015 pjT3 P2671 求和 eazy+

- [Link](#)

题目描述

[展开](#)

一条狭长的纸带被均匀划分出了 n 个格子，格子编号从1到 n 。每个格子上都染了一种颜色 $color_i$ （用 $[1, m]$ 中的一个整数表示），并且写了一个数字 $number_i$ 。



定义一种特殊的三元组： (x, y, z) ，其中 x, y, z 都代表纸带上格子的编号，这里的三元组要求满足以下两个条件：

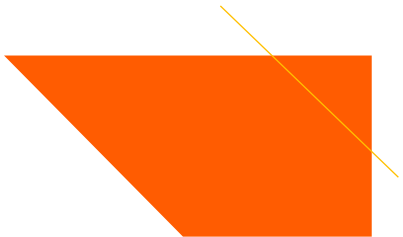
1. xyz 是整数, $x < y < z, y - x = z - y$
2. $color_x = color_z$

满足上述条件的三元组的分数规定为 $(x + z) \times (number_x + number_z)$ 。整个纸带的分数规定为所有满足条件的三元组的分数的和。这个分数可能会很大，你只要输出整个纸带的分数除以10,007所得的余数即可。



2016D2T2 P2827 蚯蚓

[Link](#)



UVA1619

[Link](#)

题目大意：给出正整数 n 和一个($1 \leq n \leq 100\,000$)长度的数列，要求找出一个子区间，使这个子区间的数字和乘上子区间中的最小值最大。输出这个最大值与区间的两个端点。



CSP-S2021 P7913 廊桥分配

[Link](#)

题目描述

[展开](#)

当一架飞机抵达机场时，可以停靠在航站楼旁的廊桥，也可以停靠在位于机场边缘的远机位。乘客一般更期待停靠在廊桥，因为这样省去了坐摆渡车前往航站楼的周折。然而，因为廊桥的数量有限，所以这样的愿望不总是能实现。

机场分为国内区和国际区，国内航班飞机只能停靠在国内区，国际航班飞机只能停靠在国际区。一部分廊桥属于国内区，其余的廊桥属于国际区。

L 市新建了一座机场，一共有 n 个廊桥。该机场决定，廊桥的使用遵循“先到先得”的原则，即每架飞机抵达后，如果相应的区（国内/国际）还有空闲的廊桥，就停靠在廊桥，否则停靠在远机位（假设远机位的数量充足）。该机场只有一条跑道，因此不存在两架飞机同时抵达的情况。

现给定未来一段时间飞机的抵达、离开时刻，请你负责将 n 个廊桥分配给国内区和国际区，使停靠廊桥的飞机数量最多。



P1886 滑动窗口

- 有一个长为 n 的序列 a ，以及一个大小为 k 的窗口。现在这个从左边开始向右滑动，每次滑动一个单位，求出每次滑动后窗口中的最大值和最小值。
- $n \leq 1e6$



Thank You