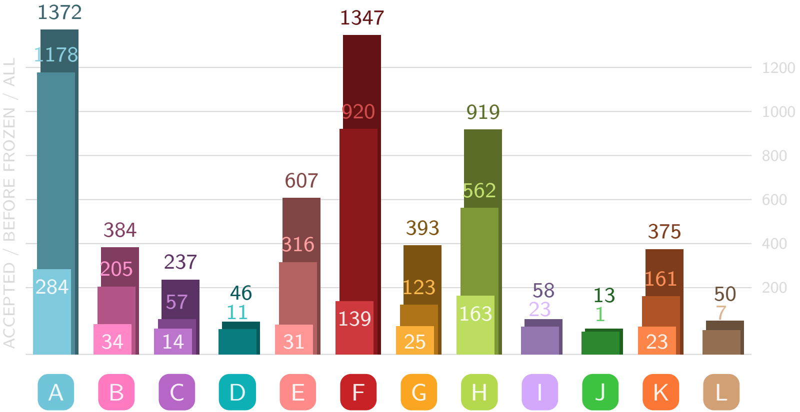
CCPC Henan Provincial Contest 2023 Solution

May 7, 2023

统计


ACCEPTED / BEFORE FROZEN / ALL
1372
1178
284
384
205
34
237
57
14
46
11
607
316
31
1347
920
139
393
123
25
919
562
163
58
23
13
1
375
161
23
50
7
1200
1000
800
600
400
200
A
B
C
D
E
F
G
H
I
J
K
L

小水獭游河南

$|a| \leq |\Sigma| = 26$，暴力枚举 a 判断 b 是否为是回文串即可，时间复杂度 $\mathcal{O}(|\Sigma||s|)$。

Art for Rest

记

$$premax_i = \max_{1 \le j \le i} a_i, \quad sufmin_i = \min_{i \le j \le n} a_i$$

对于 $1 \le k \le n$，A'_k 单调不减等价于 $premax_{ik} \le sufmin_{ik+1}$ 对所有 $1 \le i < \lceil \frac{n}{k} \rceil$ 成立，直接判断即可。

$\mathcal{O}(n \log n)$

Toxel 与随机数生成器

由于本题具有随机性，很多算法都可能通过。一种做法是，注意到错误代码的结果中一定会出现非常长的子串（长度至少为 10^3）与字符串的前缀相同。这由字符串的生成方式可以很容易发现。而正确代码产生的字符串完全随机，出现这种情况的概率仅略高于 2^{-1000}，可以认为不可能发生。为找出这样的子串，可以想到 KMP 算法，其所求得的 `fail` 数组正是我们需要的。如果 `fail` 数组的最大值高于 50，就已有极大把握认为是错误代码的结果了。

时间复杂度 $\mathcal{O}(|s|)$。

Toxel 与多彩的宝可梦世界

Identifying Common Connected Components of Graphs 论文题！

考虑 $[1,n]\cap\mathbb{Z}$ 的一个划分，要求它的每个元素在 k 个图中都是连通的。若它的任意几个元素都不能组合成一个更大的连通块，称其为极大的。

Toxel 与多彩的宝可梦世界

下面证明极大的划分是唯一的。若有两个不同的划分 S, T 都是极大的，不妨设 S 中某个元素与 T 中多个元素有交集。S 中的这个元素是在 k 个图中是连通的，所以对于 T 中这几个元素，在 k 个图中都必然能够找到一些边将它们连通起来，这与 T 极大矛盾。

Toxel 与多彩的宝可梦世界

首先注意到，只要某个点在其中一个图是孤立点，那么它最终就必须是孤立的。所以可以首先把这样的点和相关的边删掉。剩下的点，在每个图中至少有一条邻边，那么就有 $kn \le 2m$。考虑进行若干轮的拆图工作。每一轮先把每个图的所有连通块找出来，然后可以找到 k 个图的所有公共连通块。这可以将每个点的 k 个 root 基数排序，在 $\mathcal{O}(kn)$ 之内解决。

Toxel 与多彩的宝可梦世界

如果所有图的连通状态相同，这就已经是一个极大的划分，那么可以结束迭代。答案就是最大的连通块。否则，枚举所有的边，如果它连接了两个不同连通块的点，那么这条边必然不会在最终的答案里，可以删去。一轮操作之后，划分必然变得更细，所以总共最多有 $n-1$ 轮，因此总复杂度为 $\mathcal{O}(n\sum m_i)$。

矩阵游戏

考虑 $f_{i,j,k}$ 表示从 $(1,1)$ 开始走到 (i,j) 恰好替换了 k 个 ? 最多获得的分数，容易得到转移方程：

$$f_{i,j,k} = \begin{cases} \max(f_{i,j-1,k}, f_{i-1,j,k}) & s_{i,j} = \mathsf{0} \\ \max(f_{i,j-1,k}, f_{i-1,j,k}) + 1 & s_{i,j} = \mathsf{1} \\ \max(f_{i,j-1,k}, f_{i-1,j,k}, f_{i-1,j,k-1} + 1, f_{i,j-1,k-1} + 1) & k \neq 0 \land s_{i,j} = \mathsf{?} \\ \max(f_{i,j-1,k}, f_{i-1,j,k}) & k = 0 \land s_{i,j} = \mathsf{?} \end{cases}$$

矩阵游戏

其中 $s_{i,j}$ 表示 (i,j) 的字符，转移复杂度为 $\mathcal{O}(nmx)$。但是空间复杂度也为 $\mathcal{O}(nmx)$，考虑到转移过程中只有不超过 $\mathcal{O}(m)$ 个状态有意义，因此我们可以只保存每列最后更新的状态即可，转移方程只需要忽略第一维即可，空间复杂度降到 $\mathcal{O}(mx)$。

Art for Last

不妨记 A 升序排序后得到 A'。我们有以下结论：

Theorem

存在一种最优的选择方案，满足所选的项在 A' 中是连续的。

Art for Last

Proof.

若所选的项 $A'_{p'_1}, \ldots, A'_{p'_k}$ $(1 \le p'_1 < \cdots < p'_k \le n)$ 在 A' 中不连续，对于取得 $\min_{1 \le i < j \le k} \left\{ A'_{p'_j} - A'_{p'_i} \right\}$ 最小的 p'_i, p'_j，选择 $A'_{p'_j - j + 1}, \ldots, A'_{p'_j - j + k}$ 不会更劣。

这是由于 $p'_i \le p'_j - 1 \le p'_j$ 且 $p'_1 \le p'_j - j + 1 \le p'_j - j + k \le p'_k$，从而 min 项和 max 项均不大于原选择方案。 □

Art for Last

将 A 排序后，枚举所选区间的端点，单调队列维护区间中差分项的最小值即可得到 $\min$ 项，区间左右端点值之差即为 $\max$ 项，取 $\min \cdot \max$ 最小的区间即可。

$\mathcal{O}(n \log n)$

Toxel 与字符画

按照题意模拟即可。例如，一种实现方式是，将题面提供的各种字符画在程序中存入一个二维字符矩阵中。随后计算表达式的值，并求出该表达式所需使用的各个字符。最后根据这些字符，找到相对应的字符画，拼接在答案后即可。

Travel Begins

记 ε 为一个充分小的正实数，例如取 $\varepsilon = 114514^{-1919810}$。

$\{a_i\}$ 中包含尽可能多的 $0.5-\varepsilon$ 时 $\sum_{i=1}^{k} r(a_i)$ 取得最小值。

$\{a_i\}$ 中包含尽可能多的 0.5 时 $\sum_{i=1}^{k} r(a_i)$ 取得最大值。

计算 $0.5-\varepsilon$ 与 0.5 分别最多能有多少项，并计算其余部分四舍五入的值即可得到答案。

$\mathcal{O}(T)$

数正方形

考虑容斥，用总数减去非纯色正方形数。

容易发现，一个 2×2 的正方形是非纯色的，当且仅当其中心点被至少一个矩形的边覆盖。

因此可以减去每个矩形边覆盖的点数，但有一些点可能被覆盖了多次。由于坐标的特殊限制，可以发现，每个点最多被覆盖两次，且一定是一条竖边和一条横边的交点。

问题转化为求出交点的数量，这是一个经典的问题，可以用扫描线算法解决。由于坐标的特殊限制，只需要树状数组而不需要线段树。

时间复杂度 $\mathcal{O}(n \log n)$。

Mocha 沉迷电子游戏

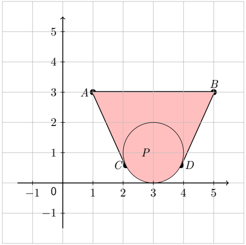
以速度 v 移动时间 t，最后距离起点最远距离为 $v \times t$。显然以起点为圆心，$v \times t$ 为半径的圆内都是危险区域。

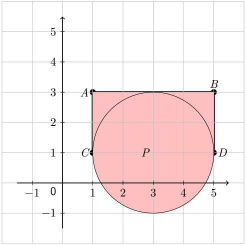
考虑危险区域的其他边界，即考虑点 Q 在圆周上移动时三角形 QAB 的变化情况，显然危险区域的边界除部分圆周外，还包括和圆相切时的线段 QA 和 QB。因此分三种情况对危险区域计算面积即可：

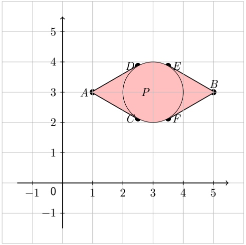
- 线段 AB 和圆相离
- 线段 AB 和圆相切
- 线段 AB 和圆相交

Mocha 沉迷电子游戏

三种情况的危险区域如下图所示：







对于每种情况分解成若干简单几何图形计算面积即可。

排列与质数

这题的定位是一个中等难度的构造题，期望达到只要花了足够的时间的队伍都能通过的效果。

事实上，验题的情况比较符合预期，也看到了各种各样不同的做法，这里提供一种几乎不需要分类讨论且实现起来比较简单的做法。

排列与质数

对于 $n \le 10$，可以暴力枚举排列求解；

对于 $n > 10$ 的奇数，先将数按照 $1, 3, 5, \ldots, n-2, n, n-3, n-5, \ldots, 8, 6, 4$ 排列；

对于 $n > 10$ 的偶数，先将数按照 $1, 3, 5, \ldots, n-3, n, n-2, n-4, \ldots, 8, 6, 4$ 排列；

即先将奇数升序排列，再将偶数降序排列。

排列与质数

可以发现，现在除了 2 和 $n-1$ 以外，所有数均已出现，且满足题目的限制。那么我们只需要将这两个数插进合适的位置即可。容易发现一定有解，因为可以将 2 插在 5 和 7 之间，将 $n-1$ 插在 $n-4$ 和 $n-6$ 之间。

复杂度取决于判断质数的速度，$\mathcal{O}(n\sqrt{n})$ 已经足以通过此题。

猜数游戏

设 $N=10^9$，交互器的既约分数为 $\frac{p}{q}$，询问的既约分数为 $\frac{a}{b}$，则

$$\frac{c}{d}=\left|\frac{p}{q}-\frac{a}{b}\right|=\frac{|pb-aq|}{bq}$$

$$c=\frac{|pb-aq|}{\gcd(|pb-aq|,bq)},\quad d=\frac{bq}{\gcd(|pb-aq|,bq)}$$

交互器返回值即为

$$r=c+d=\frac{|pb-aq|+bq}{\gcd(|pb-aq|,bq)}$$

猜数游戏

我们希望提问的 a, b 足够简单并且 $\gcd(|pb-aq|, bq)=1$。

一个可行的方案是提问 $\frac{1}{P}$，其中 P 为随机大质数，不妨设 $P \geq 0.9N$。

此时有

$$c=\frac{|pP-q|}{\gcd(|pP-q|, Pq)}, \quad d=\frac{Pq}{\gcd(|pP-q|, Pq)}$$

特殊地，返回值为 0 时可以直接确定答案。

猜数游戏

Case 1:

当且仅当 $q = P$ 时有 $\gcd(|pP - q|, Pq) = P \neq 1$，返回值为 $r = c + d = p - 1 + P \leq 2P - 2$。

Case 2:

当 $q \neq P$ 时 $\gcd(|pP - q|, Pq) = 1$，此时有

$$c = |pP - q|, \quad d = Pq$$

$pP - q$ 的正负对应 $\frac{p}{q}$ 与 $\frac{1}{P}$ 的大小关系，对其分类讨论。

猜数游戏

Case 2.1: $c = pP - q \iff \frac{p}{q} > \frac{1}{P}$

- 此时返回值 $r = pP - q + Pq = (p+q)P - q$
- $q_1 = P - (r \bmod P)$ 或 $q_2 = 2P - (r \bmod P)$
- 若 q_1, q_2 均合法则 $q_1 < N - P < 0.1N$，由

$$(p_1 + q_1)P - q_1 \leq 0.2NP < P^2 \leq (p_2 + q_2)P - q_2$$

  导出矛盾，此情况至多有一组解
- 返回值满足 $r \geq 2P - 1$

猜数游戏

Case 2.2: $c = q - pP \iff \frac{p}{q} < \frac{1}{P}$

- 此时返回值 $r = q - pP + Pq = (q-p)P + q$
- $p = 1, q = r \bmod P + P$
- 此情况至多有一组解
- 返回值满足 $(P+1)^2 - P \le r \le N(P+1) - P$

猜数游戏

接下来讨论返回值在 Case 2.1 与 Case 2.2 中均有解的情况。

Case 2.2 中返回值 $r=q_2(P+1)-P$ 满足 $r\equiv 1 \pmod{P+1}$。

若返回值在 Case 2.1 中也存在解，则 $r=(p_1+q_1)P-q_1$ 也满足 Case 2.2 的条件，从而有 $p_1+2q_1\equiv P \pmod{P+1}$ 且 $(P+1)^2-P\le r\le N(P+1)-P$。

计算可得当 $p_1+2q_1=2P+1$ 且 $P<q_2=2P-q_1\le N$ 时才可能会有多解。

猜数游戏

由于 P 随机，出现 $p_1 + 2q_1 = 2P + 1$ 的概率最坏情况下为

$$\frac{1}{\pi(N) - \pi(0.9N)} \approx \frac{\ln N}{0.1N} \approx 2.17 \times 10^{-7}$$

也即在 Case 2.1 与 Case 2.2 中均有解时 Case 2.2 给出的解出现概率更高。

猜数游戏

Case 1 与 Case 2 不可能同时有解。

在 Case 2.1 与 Case 2.2 均有解的情况下选择 Case 2.2 给出的解。

可以采用区间筛、Miller Rabin 检查随机数等方式获得足够多的随机大质数。