

A.ABB

在一个字符串 str 末尾添加尽可能少的字符，使得这个字符串成为回文串。

通过马拉车或者回文树得到以末尾字符为结尾的最长回文子串 s

则答案为 $len(str) - len(s)$

B.Be Geeks

通过单调栈求出每个 a_i 作为最大值管辖的区间 $[l_i, r_i]$ 。

则每个位置的贡献为 $a_i * \sum \sum G(l, r), (l_i \leq l \leq i, i \leq r \leq r_i)$

设 $F(i, j)$ 代表 i 到 j 这段区间内所有子区间的 gcd 之和

第二段可以求和的数值可以转换为 $F(l_i, r_i) - F(l_i, i - 1) - F(i + 1, r_i)$

至此，问题还剩下给出一个序列，多次询问一个区间的所有子区间 gcd 之和，这是一个经典问题。

从一个空集合开始，不断添加一个数字并求 gcd ，那么中途所有的 gcd 出现个数是 \log 级别的。因为 gcd 每次变化是递减并且至少除以2的。

那么我们可以通过线段树上二分求出每个位置向右的所有子区间的 gcd 值，分为 \log 段存在 $vector$ 里。

建线段树，每个叶子维护以这个点为右端点的子区间 gcd 。

接下来将询问离线，按照左端点降序排序，然后从右往左将序列中的各值加入贡献，于是问题变成区间修改区间求和，线段树足以胜任。

如果使用了线段树上二分时间复杂度足以优化到两个 \log ，但此题的 std 是三个 \log 的，因此其他合适的做法也可以通过。

C.Bob in Wonderland

贪心，统计有多少个度大于2的节点数 x ，将他们的度求和 sum ，答案为

$sum - 2x$

D.Deep800080

求出直线与所有圆的交线段，问题变成有许多个区间，问最多被覆盖的一段的被覆盖次数。这个问题可以直接差分，然后枚举每个值取最大即可。

E.Zeldain Garden

洛谷P3952

答案可以进行做差，那么对于某一个值 i ，只需要求 $\sum d(x), 1 \leq x \leq i$ ，可以通过枚举约数然后分块解决。

F.Light Emitting Hinderburg

从 n 个数中选 k 个数，让 k 个数的按位与运算最大。

显然我们可以按位贪心，从高位向低位枚举。若当前位为1的数字的数量大于等于 k 个，那我们就把这一位选1，把所有这一位不是1的数全部去掉，然后继续枚举下一位；否则这一位选0，继续枚举下一位。

复杂度为 $O(N \log MAX)$ 。

G.K==S [Kiss]

- 建 AC 自动机。

- AC 自动机的一些点，会因为“被禁止的子串”而不可达，删除掉这些点。
- 问题转化为，在一个图上确定起点求走 n 步的方案。矩阵快速幂可以 $O(L^3 \log N)$ 地实现，其中 L 为 AC 自动机上节点个数。

H.Ponk Warshall

- $s[i]$ 向字符 $t[i]$ 连有向边。
- 问题转化为把边集划分成尽可能多的环。
- 贪心地拿长度为 2 的环，长度为 3 的环，长度为 4 的环。

I.Saba1000kg

观察可得：所有子图中的边数之和最多为 $O(N\sqrt{N})$

证明：最坏的情况下，一个查询子图中包含 \sqrt{N} 个点和与 N 相同数量级的边，最多有 \sqrt{N} 个这样的查询。

考虑查询的两种情况：

- 对于 M 大于 \sqrt{N} 的查询，遍历所有边并检查它们是否在给定的子图中。这样的查询次数不会超过 \sqrt{N} 次，因此访问的边总数不超过 $N\sqrt{N}$ 次。
- 对于 M 小于等于 \sqrt{N} 的查询，枚举所有顶点对，检查是否有连接它们的边。访问的边总数不超过 $\Sigma M^2 \leq N\sqrt{N}$ 次。检查可以用 $O(\log N)$ 的时间完成（例如使用 set）

查询子图的连通块的数量可以通过若干次 DFS 在线性时间内完成。总时间复杂度为 $N\sqrt{N}O(\log N)$

K.The Bugs

题意

给定一个序列 a_i ，共 n 个数， $n \leq 2 \times 10^5, 1 \leq a[i] \leq 10^9$ 。

现在取出一个大小为 3 的序列（保留原顺序），组成一个三元组 (x, y, z) ，定义其特征三元组为 $(sgn(y-x), sgn(z-y), sgn(z-x))$ 。

其中 $sgn(x)$ 定义为：

$$sgn(x) = \begin{cases} -1 & x < 0 \\ 0 & x = 0 \\ 1 & x > 0 \end{cases}$$

要求找出所有的可能出现的特征三元组，可以生成该特征三元组的所有可能的三元组，每一个数都大于 0 并且字典序最小的那样一个。最终结果按字典序输出。

题解

显然可以发现，可能的情况只有 13 种，分别是

111, 112, 121, 211, 122, 212, 221, 123, 132, 213, 231, 312, 321。

分别判断输出即可。

接下来举四个具有代表性例子：

111：判断是否一个数出现了大于等于 3 次即可。

121：对于每一个数，与之相等的最靠前的和最靠后的两个位置。判断两个位置之间的最大值是否大于该数即可。查询区间最大值可以预处理 ST 表。

123：枚举每一个位置的数，查询该位置之前的小于该数的最大值，以及该位置之后的大于该数的最小值，判断大小即可。这里提供一种方法：维护两个 *multiset*， L 和 R 。一开始将所有的数压入 R 。然后从一个数开始扫，每扫过一个数，就把这个数从 R 中删去，然后加到 L 里。如果要查询的话，直接在两个 *multiset* 里 *upper_bound* 和 *lower_bound* 即可。

132：枚举每一个位置的数，查询该位置之前的最小值，以及该位置之后的小于该数的最大值，判断大小即可。判断方法同 123。