A FibTree

- 首先我们容易发现所有生成树的权值和f(T)处于一个区间[L,R]中，L=f(最小生成树)，R=f(最大生成树)，且每一个权值和都能取到，证明留给读者思考。
- 其次我们发现斐波那契数增长得很快，在有限的范围内只有29个。
- 每次暴力判断每个斐波那契数是否落在对应区间[L,R]中。
- 时间上可以使用Kruskal和堆优化的Prim，时间复杂度分别是O(mlogm)和O(mlogn)。
- 注意细节：重边无需特殊处理，自环忽略。
- 扩展版：权值不是0或者1，而是任意数字，怎么做？欢迎与我交流你的想法！

B：Strings

题意

给定一个长度为 10^5、字符集为大写字母的字符串 S。

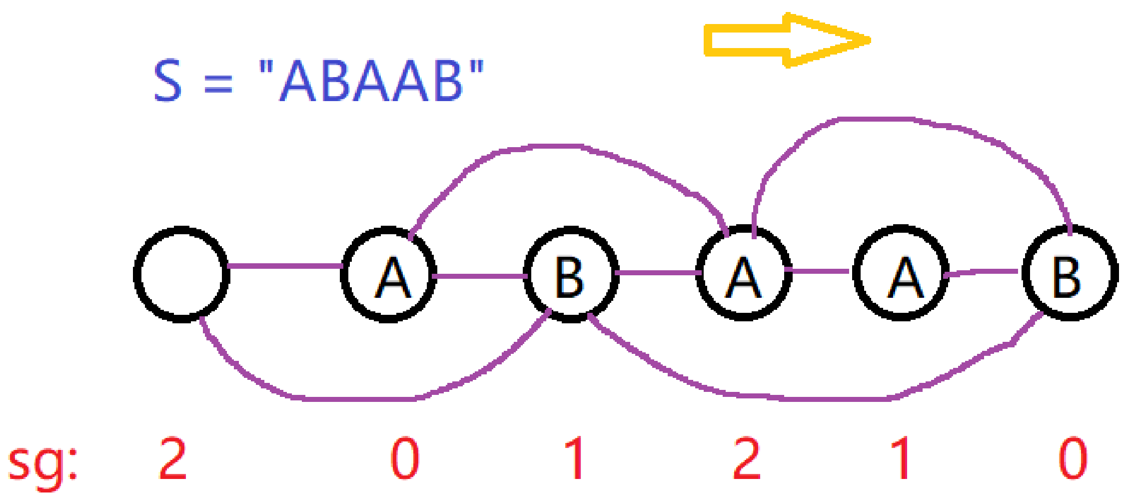
小 Z 和小 Y 玩游戏，轮流操作，小 Z 先手，谁无法操作谁输。

每个回合，都有一个由 S 的若干个子序列构成的集合 T。最初，T 中仅含一个空字符串。

每个回合内的操作规则：操作方要选择一个 T 中的字符串 s，将 s 从 T 中删除，并将至少一个形如 $s+c$、且 $s+c$ 仍然为 S 的子序列的字符串加入 T 中。

分析

先构建子序列自动机，以 S = ABAAB 为例：



规则：最初，空白点有一个石子。每一回合，操作一方可以选择一个石子，将其移除，并在其后继中放至少一个石子。

记 $sg(i)$ 表示最初点 i 上放一个石子的局面的 SG 值，答案即为 $sg(0)$。

$$sg(x) = mex_{T\subset succ(x)}(\otimes_{k\in T}sg(k))$$

容易发现 SG 值只可能是 $0, 2^0, 2^1, 2^2, \dots$

按拓扑序逆序归纳：

- 对没有后继的点，$sg=0$
- 对点 x，其后继的 sg 值若没有 0，其 sg 值为 0，否则假设 $2^0, 2^1, \dots, 2^i$ 都存在，2^{i+1} 不存在，那么 2^{i+1} 是它不能通过子集异或得到的最小的数，所以 $sg=2^{i+1}$

将其重标号为 $0, 1, 2, 3, \dots$，再来考虑上述求 SG 的方法，每次相当于对所有后继的重标号取 mex，**并特殊考虑有两个数 mex 值相同的情况。**

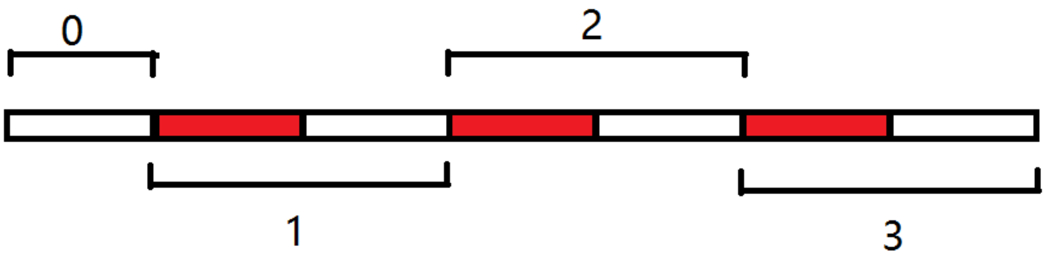
C – More and More:

题意即为从n个数中选取最多m个不相交的子段，将其中所有数$*$ x后求新数组的最大子段和

考虑最终选取的子段，一定是这样的形式：

其中红色表示被改变的数

那么我们可以将这些数分组，每一组包含两部分：被改变的和未改变的，注意第0组只有未改变的



记f[i][j]表示以 第i个数处于第j组中红色部分 结尾的最大子段和

记g[i][j]表示以 第i个数处于第j组中白色部分 结尾的最大子段和

容易得到

$$f[i][j] = \max(f[i-1][j], g[i-1][j-1]) + a[i] * x, j > 0$$
$$g[i][j] = \max(f[i-1][j], g[i-1][j]) + a[i]$$

但是，由于可能会有$a[i] < 0$导致$f < 0$或者$g < 0$，因此还要加上

$$f[i][j] = \max(f[i][j], a[i] * x), j > 0$$
$$g[i][j] = \max(g[i][j], a[i])$$

表示放弃之前的重新开始

这样写虽然会导致将新的选择方案的第一个区间记录为j而不是0造成区间的浪费，但并不会影响最终答案

D Sakuya’s task

(~~杜教筛人口普查~~)

- 这次不小心把题目出简单了（我自裁）
- 原本做法：（原本是n!=m的）
- $\sum_{i=1}^{n}\sum_{j=1}^{m}\varphi(\gcd(i,j)) = \sum_{p=1}^{\min(n,m)}\varphi(p)\sum_{i=1}^{n}\sum_{j=1}^{m}[\gcd(i,j)==p] = \sum_{p=1}^{\min(n,m)}\varphi(p)\sum_{d=1}^{\min(n/p,m/p)}\mu(d)\left\lfloor\frac{n}{pd}\right\rfloor\left\lfloor\frac{m}{pd}\right\rfloor = \sum_{T=1}^{\min(n,m)}\left\lfloor\frac{n}{T}\right\rfloor\left\lfloor\frac{m}{T}\right\rfloor\sum_{p|T}\varphi(p)\mu\left(\frac{T}{p}\right)$
- 先数论分块，这时候需要求f(T)=φ*μ的前缀和，用两次杜教筛均卷上恒为1的函数即可

- 中学生做法：
- $\sum_{i=1}^{n}\sum_{j=1}^{n}\varphi(\gcd(i,j)) = \sum_{p=1}^{n}\varphi(p)\sum_{i=1}^{n}\sum_{j=1}^{n}[\gcd(i,j) == p] = \sum_{p=1}^{n}\varphi(p)\left(2\sum_{i=1}^{n/p}\varphi(i) - 1\right) = \sum_{p=1}^{n}\varphi(p)\left(2S\left(\left[\frac{n}{p}\right]\right) - 1\right)$
- 经典数论分块+杜教筛

E Election

- 设$f(i)$为以i为根的树上逆序对之和，求$\sum f(i)$

BruteForce

Sol1

- 纯暴力
- $O(n^3)$

Sol2

- 权值不同的(i,j)对答案的贡献是$n-dis(i,j)+1$，求LCA即可
- $O(n^2)$

正解1

- 首先考虑求$f(1)$，对于每个点i，对$f(1)$的贡献是其子树内权值小于它的点的个数，加起来即可
- 采用DFS统计答案，从i转移到j时，$f(j)=f(i)-lf+ls$，其中lf为以j为根的子树内权值小于a_i的点数，ls为除了以j为根的子树外权值小于a_j的点数

Sol3

- 在DFS序上对所有点权建立可持久化线段树
- $O(nlogn)$

Sol4

- DSU + 树状数组
- $O(nlog^2n)$

Sol5

- LCT
- $O(nlogn)$

正解2

- 将点权排序，依次考虑每个点对答案的贡献

Sol6

- 在DFS序上建立树状数组，枚举到点i时，将小于a_i的点扔进树状数组中
- 枚举从点i出发的每一条边，如果是son，对答案的贡献为$(n-size[son])*Ask(son)$，如果是$father$，对答案的贡献是$size[i]*(Ask(all)-Ask(i))$
- $O(nlogn)$

Sol7

- 将点权序上建立树状数组，以1为根时，将每个点的子树大小扔进树状数组中，$Ask(all) - Ask(a_1)$就是1对答案的贡献
- DFS统计答案，从i到son时，为了维护树状数组，将点i的子树大小$-size[i] + n - size[son]$即可
- $O(nlogn)$

F Charmander and His Best Friends:

考虑构造出这样的解：不仅满足题意，还满足左边一半的最大值<右边一半的最小值。

假如我们要求解的长度为 n:

假设我们已经求出满足条件的长度为 n/2（所有分式为下取整）的数组 b 和长度为 n-n/2 的数组 c，将它们按这样的方式拼接：

$$b_{\left\lfloor\frac{n}{2}\right\rfloor},\ldots,b_1,c_{n-\left\lfloor\frac{n}{2}\right\rfloor}+\left\lfloor\frac{n}{2}\right\rfloor,\ldots,c_1+\left\lfloor\frac{n}{2}\right\rfloor$$

就得到了长度为 n 的解。(如果把它看成四部分, c 的左边>c 的右边>b 的左边>b 的右边）所以任意跨过中间的限制一定能被满足。