3.13纪中集训（联考5）

标签（空格分隔）： 2019集训

T1 碱基配对

一道原题：Codeforces528D fuzzy search

T2 小凯的疑惑

有n个点，每个点有一个点权v_i，保证$v_i < 2^C$。任意两个点之间都有边，边权是两个点的点权的异或。每一次操作给出x，把所有的点的v_i变成$v_i + x \mod 2^C$。你需要在每一次操作过后，输出最小生成树的边权和。操作有后效性。$n, q \le 20000, C \le 14$

Solution：用一个长度为2^C的01序列表示每一个数是否作为n个点的点权出现过，则每一次操作相当于把这个序列向右循环移动x位。我们考虑直接计算出原来的序列以第x位作为修改后的01序列的第一个元素，得到的最小生成树的权值。

按照求最小异或生成树的套路，我们对出现过的点权建trie。暴力计算的时候，复杂度集中在计算两棵子树内选一个点，求最小异或值的这个过程。于是我们考虑下面的两个~~蛇皮的~~优化

- 我们先处理出高度为3的所有可能的子树，两两之间查询最小异或值的结果。高度为3的子树可能有2^{2^3}种，复杂度可以接受。
- 对于高度大于3的子树，假设高度为d，我们处理出$[i, i + 2^d)$和$[i + k \cdot 2^d, i + (k+1) \cdot 2^k)$这些子树（也就是左端点在模$2^d$的意义下是相同的）两两之间的最小异或值。显然每一棵子树只会与2^{C-d}棵子树进行计算。由于我们已经处理完了高度为$d-1$的所有的子树两两之间的最小异或值，所以重新计算两棵子树的复杂度是$O(1)$的，而这一层的复杂度等于这一层我们需要计算的子树的对数，即$2^C \cdot 2^{C-d}$，当$d = 4, C = 14$时，复杂度是$2^{14} \cdot 2^{10}$，可以通过。

T3 false-false-true (fft)

有$n+m$道题，其中有n道题的答案是yes，m道题的答案是no。小z并不知道哪些题是yes，哪些题是no，但是他知道n和m，并且在他答完一道题之后，他会知道自己是答对了还是答错了。输入n,m，问在小z采取最优策略的前提下，他答错的题目的数量期望。
$n,m\le 500000$

Solution：我觉得$n=m\le 10^5$的暴力分挺有意思的。

设$f_{i,j}$表示仍然剩下i道yes和j道no的时候，期望答错的题数。那么
$f_{i,j}=\frac{i}{i+j}f_{i-1,j}+\frac{j}{i+j}f_{i,j-1}+\frac{\min(i,j)}{i+j}$。考虑每一个$(i,j)$的$\frac{\min(i,j)}{i+j}$对最终答案的贡献，可以推出：

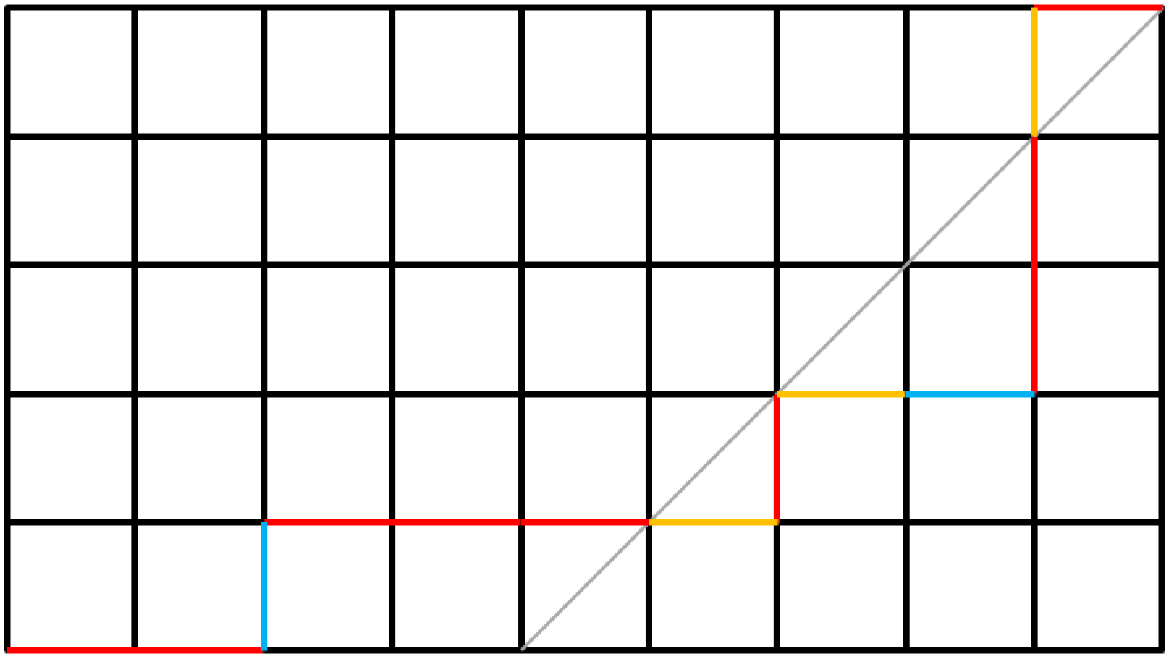
$$Ans=\sum_{i=1}^{n}\sum_{j=1}^{m}\frac{\min(i,j)}{i+j}\frac{n!}{i!}\frac{m!}{j!}\frac{(i+j)!}{(n+m)!}\binom{n-i+m-j}{n-i}$$
$$=\sum_{i=1}^{n}\sum_{j=1}^{m}\frac{n!m!}{(n+m)!}\frac{(n+m-i-j)!(i+j)!}{(i+j)}\frac{1}{i!(n-i)!}\frac{1}{j!(m-j)!}\min(i,j)$$

如果没有$\min(i,j)$这样的东西就直接卷积。对于这个$\min(i,j)$的限制，再套个分治就可以了。具体来说就是加入现在是$i\in[1,n],j\in[1,n]$，那么我们先让
$i\in[1,mid),j\in[mid,r]$，这样计算的时候i,j的大小关系就确定了。还要算
$j\in[1,mid),i\in[mid,r]$。然后再递归下去算i,j都在左区间或者右区间的情况。

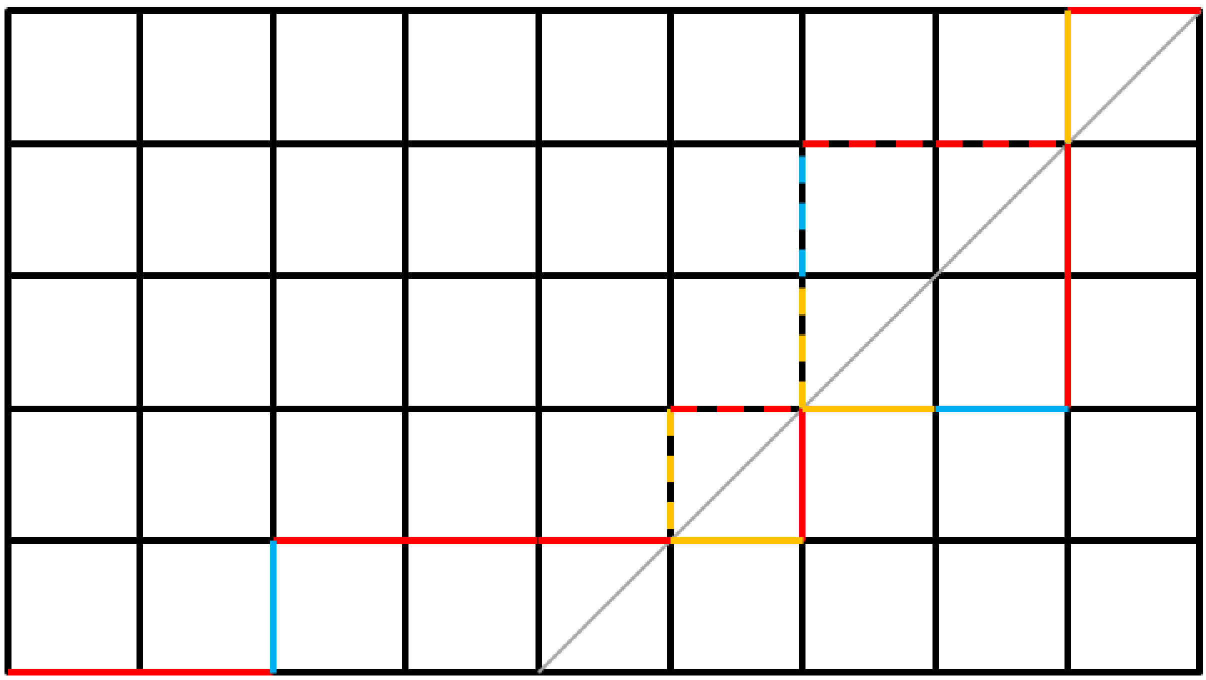
其实也就是分治的套路：枚举划分点，考虑跨越划分点的贡献，然后递归处理不跨越划分点的贡献。$\min(i,j)$就神奇地在我们枚举划分点的时候确定了。

正解很有意思：令$n>m$。我们实际上是在网格图上从(n,m)走到$(0,0)$，要求只能向左或者向下走。而在(i,j)这个格子，我们的策略是：$i>j$我们就猜yes，否则就猜no。我们画出(n,m)到$(0,n-m)$这条线，线左边的路径，我们每往右走一步就会产生1的贡献；线右边的路径，我们每往上走一步就会产生1的贡献；而路径中在线上的部分，产生的贡献是$\frac{1}{2}$。通过翻折发现线左边向右的步数 + 线右边向上的步数 = n。于是我们只需要算线上的部分的贡献，枚举线上的每一个点，算一算经过它的概率就可以了。

下面是一个例子。$n=9,m=5$。红色的线贡献为1，蓝色的线的贡献为0，黄色的线的贡献是$\frac{1}{2}$。



我们把灰色线右边路径按照灰色线翻折：



可以发现，贡献为1的线的数量恰好等于n，贡献为$\frac{1}{2}$的线的数量等于这条路径与灰色的线的交点个数。