水

yyc

January, 2018

Codeforces 875E

有一个长度为n的数列，由$n-2$个x和2个y组成，你可以询问不超过19次，每次可以问一些数的异或和，要求找出两个数值为y的位置。

$n \leq 1000, 1 \leq x, y \leq 10^9, x \neq y$。

我们的询问只会得到四种值：$0, x, y, x\ xor\ y$。

很显然，如果我们询问到了一个集合的异或和为y或$x\ xor\ y$，那么就说明这个集合有且仅有一个y，可以直接在这个集合里面二分找出。

我们的询问只会得到四种值：$0, x, y, x\ xor\ y$。

很显然，如果我们询问到了一个集合的异或和为y或$x\ xor\ y$，那么就说明这个集合有且仅有一个y，可以直接在这个集合里面二分找出。

设数值为y的位置为$p_1, p_2 (p_1 < p_2)$，我们就需要找出一个集合只包含$p1$或$p2$。

我们的询问只会得到四种值：$0, x, y, x\ xor\ y$。

很显然，如果我们询问到了一个集合的异或和为y或$x\ xor\ y$，那么就说明这个集合有且仅有一个y，可以直接在这个集合里面二分找出。

设数值为y的位置为$p_1, p_2(p_1 < p_2)$，我们就需要找出一个集合只包含$p1$或$p2$。

由于$p_1 \neq p_2$，所以$p_1\ xor\ p_2 > 0$，所以可以对于每个二进制位，询问所有这一位为1的位置，总会询问出y或$x\ xor\ y$，那么在这个集合里面二分，就可以找到其中一个值了。

我们的询问只会得到四种值：$0, x, y, x\ xor\ y$。

很显然，如果我们询问到了一个集合的异或和为y或$x\ xor\ y$，那么就说明这个集合有且仅有一个y，可以直接在这个集合里面二分找出。

设数值为y的位置为$p_1, p_2 (p_1 < p_2)$，我们就需要找出一个集合只包含$p1$或$p2$。

由于$p_1 \neq p_2$，所以$p_1\ xor\ p_2 > 0$，所以可以对于每个二进制位，询问所有这一位为1的位置，总会询问出y或$x\ xor\ y$，那么在这个集合里面二分，就可以找到其中一个值了。

另一个位置其实也很好求，只要某个二进制位询问出来是y或$x\ xor\ y$，那么p_1和p_2这一位就会不同，所以直接异或一下就行了。

我们的询问只会得到四种值：$0, x, y, x\ xor\ y$。

很显然，如果我们询问到了一个集合的异或和为y或$x\ xor\ y$，那么就说明这个集合有且仅有一个y，可以直接在这个集合里面二分找出。

设数值为y的位置为$p_1, p_2(p_1 < p_2)$，我们就需要找出一个集合只包含$p1$或$p2$。

由于$p_1 \neq p_2$，所以$p_1\ xor\ p_2 > 0$，所以可以对于每个二进制位，询问所有这一位为1的位置，总会询问出y或$x\ xor\ y$，那么在这个集合里面二分，就可以找到其中一个值了。

另一个位置其实也很好求，只要某个二进制位询问出来是y或$x\ xor\ y$，那么p_1和p_2这一位就会不同，所以直接异或一下就行了。

总询问次数为$2 * log_2(n) + 1 = 19$次。

有n个嫌疑人，总共说了m条供词。

供词有以下两种形式：

$t = 0 : x$说y是犯人；

$t = 1 : x$说y不是犯人。

其中有若干个犯人，他们每个人的供词最多有一句是假的，其他人的供词总是真的。

你需要找出哪些人是犯人（可能多解或无解，多解求出一组即可）。

$n, m \leq 10^5$。

这题容易想到2-sat，接下来我们考虑如何建模。

这题容易想到2-sat，接下来我们考虑如何建模。
对于每个人和每条供词都建立一对点。
i_0表示第i个人是犯人或第i条供词是假的。
那么对于每个人和他说的供词就可以存在着一些关系：
$i_1 -> \ j_1, j_0 -> \ i_0$。

这题容易想到2-sat，接下来我们考虑如何建模。
对于每个人和每条供词都建立一对点。
i_0表示第i个人是犯人或第i条供词是假的。
那么对于每个人和他说的供词就可以存在着一些关系：
$i_1 -> \ j_1, j_0 -> \ i_0$。
再看供词的内容，若被描述的嫌疑人为y，则有这样一些关系：

这题容易想到2-sat，接下来我们考虑如何建模。
对于每个人和每条供词都建立一对点。
i_0表示第i个人是犯人或第i条供词是假的。
那么对于每个人和他说的供词就可以存在着一些关系：
$i_1 -> \ j_1, j_0 -> \ i_0$。
再看供词的内容，若被描述的嫌疑人为y，则有这样一些关系：
若$t = 0:$
$j_1 -> \ y_0, y_1 -> \ j_0$。
若$t = 1:$
$j_1 -> \ y_1, y_0 -> \ j_0$。

另外，每个人最多说一句假供词，所以对于同一个人说的供词j,k，有以下关系：

$j_0 -> k_1, k_0 -> j_1$。

另外，每个人最多说一句假供词，所以对于同一个人说的供词j,k，有以下关系：

$j_0 -> k_1, k_0 -> j_1$。

至此，我们只需要将边直接连出来跑一遍2-sat就可以得出解了。

但是可以发现，如果一个人说的供词特别多，那么就会连m^2条边，显然是无法承受的。

那么可以对于每条供词再加上一对点，表示这个人之前说过的所有供词中是否全为真。

设J为第i个人在j之前说的所有供词的情况，k是j的下一个，则可以连这样一些边：

$J_1 -> \ k_0, k_1 -> \ J_0$

$J_1 -> \ K_1, K_0 -> \ J_0$

$J_0 -> \ j_0, j_1 -> \ J_1$

并且$i_1 -> \ j_1, j_0 -> \ i_0$的边可以不需要连了，只需连一条$i_1 -> \ J_1, J_0 -> \ i_0$($J$是$i$说的最后一句话)即可。

给出一个长度为n的字符串s，字符集为a-h（大小为8），对于每一对$i \neq j$，若$|i-j| = 1$或$s_i = s_j$，则i, j间连一条权值为1的无向边，否则不连边。

求这n个点构成的图的直径以及最短路等于直径的点对数量。

$2 \leq n \leq 10^5$。

记录$DIS(i,j)$为i,j间最短路。

记录$DIS(i,j)$为i,j间最短路。

可以先模拟一些DIS。

这里以这个串举例子：abcbde。

我们模拟一下$DIS(1,5)$和$DIS(3,6)$，并将路径上的字符记下来。

$1,5$：abbd，长度为3。

$3,6$：cbde，长度为3。

记录$DIS(i,j)$为i,j间最短路。

可以先模拟一些DIS。

这里以这个串举例子：abcbde。

我们模拟一下$DIS(1,5)$和$DIS(3,6)$，并将路径上的字符记下来。

$1,5$：abbd，长度为3。

$3,6$：cbde，长度为3。

可以发现，路径中最多出现两个相同字符，且它们必须相邻。

继续观察，可以发现，如果路径上没有两个相同字符，那么它们之间的最短路就是$|i-j|$。

继续观察，可以发现，如果路径上没有两个相同字符，那么它们之间的最短路就是$|i-j|$。

否则，可以记录$dis(i,c)$表示i到字符c的最短距离，则一定可以找到一个路径中出现了2次的c，使得$DIS(i,j)=dis(i,c)+dis(j,c)+1$。

继续观察，可以发现，如果路径上没有两个相同字符，那么它们之间的最短路就是$|i-j|$。

否则，可以记录$dis(i,c)$表示i到字符c的最短距离，则一定可以找到一个路径中出现了2次的c，使得$DIS(i,j)=dis(i,c)+dis(j,c)+1$。

为了方便，假设$i>j$。

如果i,j间最短路在第一种情况中产生，那么易证明$i-j<=7$。

那么对于每个i，可以直接枚举$j>=i-7$，

则$DIS(i,j)=min(i-j,min(dis(i,c)+dis(j,c)+1))$。

对于$j < i - 7$，$DIS(i,j) = min(dis(i,c) + dis(j,c) + 1)$。

对于$j < i - 7$，$DIS(i, j) = min(dis(i, c) + dis(j, c) + 1)$。

设$d(c, d)$为字符c, d间的最短路。

则$d(s_i, c) \leq dis(i, c) \leq d(s_i, c) + 1$。

那么可以用一个二进制数$mask_i$维护对每个c，$dis(j, c) - d(s_j, c)$的值。

对于$j < i - 7$，$DIS(i,j) = min(dis(i,c) + dis(j,c) + 1)$。

设$d(c,d)$为字符c,d间的最短路。

则$d(s_i,c) \le dis(i,c) \le d(s_i,c) + 1$。

那么可以用一个二进制数$mask_i$维护对每个c，$dis(j,c) - d(s_j,c)$的值。

记录$cnt(c,m)$表示$s_j = c, mask_j = m, j < i - 7$的数量。

那么就可以枚举c,m，直接计算答案了。

对于$j < i - 7$，$DIS(i, j) = min(dis(i, c) + dis(j, c) + 1)$。

设$d(c, d)$为字符c, d间的最短路。

则$d(s_i, c) \le dis(i, c) \le d(s_i, c) + 1$。

那么可以用一个二进制数$mask_i$维护对每个c，$dis(j, c) - d(s_j, c)$的值。

记录$cnt(c, m)$表示$s_j = c, mask_j = m, j < i - 7$的数量。

那么就可以枚举c, m，直接计算答案了。

dis数组和d数组可以对每个字符都做一次BFS求出。

时间复杂度是$O(n * sigma^2 * 2^{sigma})$;

谢谢大家！