

Menci's Blog

幻梦终醒，不悔华年

「BZOJ 2844」albus 就是要第一个出场 - 线性基

📅 2017-02-27 | 📁 OI

给定 n 个数 $\{a_i\}$ ，以及一个数 x 。将 $\{a_i\}$ 的所有子集（可以为空）的异或值从小到大排序得到序列 $\{b_i\}$ ，请问 x 在 $\{b_i\}$ 中第一次出现的下标是多少？保证 x 在 $\{b_i\}$ 中出现。

链接

[BZOJ 2844](#)

题解

首先，求出这 n 个数的线性基。

考虑线性基所控制的某个二进制位，如果这一位为 1，那么线性基中控制这一位的元素一定被选择，这样可以求出 x 在去重后的 $\{b_i\}$ 中第一次出现的下标是多少。

之后，计算每个重复的数字出现了多少次。设 a 中不在线性基中的数的集合为 S ($|S| = n - |B|$)，考虑它的一个子集 S' （可以为空）， S' 的异或和一定可以唯一表示为 B 中若干个数的异或和，将它们都异或起来，我们可以得到 0，这样，我们就得到了 $2^{n-|B|}$ 种方案得到 0，所以，对于每一个 b_i ，它的出现次数至少为 $2^{n-|B|}$ 。接着证明它的上界，假设在 S 中任意选，最终都可以凑出这个数，而选择 B 中的数的方案一定是唯一的，即上界也为 $2^{n-|B|}$ 。

代码

```
#include <cstdio>
#include <vector>

const int MAXN = 100000;
const int MAXL = 30;
const int MOD = 10086;

struct LinearBasis {
    std::vector<int> bit;
```

```
void build(int *x, int n) {
    std::vector<int> a(MAXL + 1);

    for (int i = 1; i <= n; i++) {
        int t = x[i];

        for (int j = MAXL; j >= 0; j--) {
            if (!(t & (1 << j))) continue;

            if (a[j]) t ^= a[j];
            else {
                for (int k = 0; k < j; k++) if (t & (1 << k)) t ^= a[k];
                for (int k = j + 1; k <= MAXL; k++) if (a[k] & (1 << j)) a[k] ^= t;
                a[j] = t;
                break;
            }
        }
    }

    bit.clear();
    for (int i = 0; i <= MAXL; i++) if (a[i]) bit.push_back(i);
}

int size() {
    return bit.size();
}

int rank(int x) {
    int res = 0;

```

快速幂 # BZOJ # 线性基

◀ 「WC2011」Xor - 线性基

「SCOI2016」幸运数字 - 线性基 + 树链剖分 ▶

/ 倍增

2条评论

Menci's Blog

 Disqus 隐私政策 登录 ▾ 推荐 推文 分享

评分最高 ▾



加入讨论...

通过以下方式登录

或注册一个 DISQUS 帐号 

Rapiz • 4 年前

我非常好奇这原题面是怎么能看懂的

12 ^ | v • 回复 • 分享 ▾



Menci 管理员 ➔ Rapiz • 4 年前

一脸智障.jpg

14 ^ | v • 回复 • 分享 ▾



订阅



在您的网站上使用 Disqus 添加 Disqus 添加

 不要出售我的数据

© 2015 – 2021 ♥ Menci

运行于 [GigsGigsCloud](#) 云平台 | 由 [Upyun](#) 提供 CDN 服务由 [Hexo](#) 强力驱动 | 主题 – [NexT.Pisces](#) v5.1.2