

# CF986B Petr and Permutations

[原题链接](#)

## 题目大意

给你一个有 $n$ 个元素的数组 $a$ ，让你构造一个数组 $b$ ，满足从 $a$ 、 $b$ 中任选出 $k$ 个下标对应的元素，它们的和不同，其中 $(0 < k < n)$ ，也就是说选择的子集不能是空集或全集。注意，数组中每个数都互不相同。

（来源：<https://blog.csdn.net/zuzhiang/article/details/78575126>）

## 题解

害怕，乱搞了一发，然后就A掉了（我-1都没判啊）。

大概就是排序之后全部右移一位，然后把最大的数于最小的数配。

现在想想确实是这样啊。

这里给出证明：

假设排序后数组为 $\{a_i\}$ ，右移后数组为 $\{b_i\}$ 。如果集合不选 $a_1, b_1$ ，那么显然有 $\sum a < \sum b$ 。选 $a_n, b_n$ ？把 $\{x_i\}$ 取个补集即可（两个数组的总和是相等的嘛）。

```
#include <cstdio>
#include <algorithm>

using namespace std;

const int maxn = 30;

int ans[maxn];

struct E
{
    int id, x;

    inline bool operator < (const E& other) const
    {
        return this->x < other.x;
    }
}
```

```
} aa[maxn];

int main()
{
    int n;
    scanf("%d", &n);
    for(int i = 1; i <= n; ++i)
    {
        scanf("%d", &aa[i].x);
        aa[i].id = i;
    }
    sort(aa + 1, aa + n + 1);
    for(int i = 1; i < n; ++i)
        ans[aa[i].id] = aa[i+1].x;
    ans[aa[n].id] = aa[1].x;
    for(int i = 1; i <= n; ++i)
        printf("%d ", ans[i]);
    return 0;
}
```

所以为什么 $n$ 那么小呢？大概是spj的复杂度有问题吧……