I. PERMUTATION GRAPHS

PROBLEM DESCRIPTION

給兩列 1 ~ n 的全排列，並在兩列數字相同的點間建邊，問總共會有幾個交點。

SOLUTION TECHINQUES

逆序數對

SOLUTION SKETCHES

若兩個數字 a 與 b 在兩個序列順序不一樣的話 (a 在前面或是 a 在後面)，a – a, b – b 會有交點；

所以我們可以把其中一個數列當成正確位置，去看另一個數列有哪些數字的排列不正確的，也就是求逆序數對。

TIME COMPLEXITY

$O(T * N \log N)$ T 為測資筆數，N 為數列大小。

SOLUTION PROGRAM FOR REFERENCE

```
#include <iostream>
#include <cstdio>
#include <cstring>
#include <cstdlib>
#include <algorithm>

using namespace std;
typedef long long ll;

const int N = 1e5 + 2;
const int inf = 1e9;

int a[N];
int pos[N];

int tmp[N];
ll merge_sort(int a[], int l, int r) {
    if (l == r) return 0;
    ll ret;
    int i, j, it = l, m = (l + r) >> 1;
    ret = merge_sort(a, l, m);
    ret += merge_sort(a, m + 1, r);
    for (i = l, j = m + 1; i <= m || j <= r; ) {
        int n1 = i <= m ? a[i] : inf, n2 = j <= r ? a[j] : inf;
        if (n1 < n2) {
            tmp[it++] = n1;
            ++i;
        } else {
            tmp[it++] = n2;
            ++j;
            ret += m - i + 1;
        }
    }
    for (i = l; i <= r; i++) a[i] = tmp[i];
    return ret;
}

int main() {
    int i, j, tt, n, ai;
    scanf("%d", &tt);
    while (tt--) {
        scanf("%d", &n);
        for (i = 0; i < n; i++) {
            scanf("%d", &ai);
            pos[ai] = i;
        }
        for (i = 0; i < n; i++) {
            scanf("%d", &ai);
            a[i] = pos[ai];
        }
        printf("%lld\n", merge_sort(a, 0, n - 1));
    }
    return 0;
}
```