

ROUND 2E - Thuật toán sắp xếp

Sau khi học một vài thuật toán sắp xếp trên lớp, Tí và Tèo cũng đã thử tìm những thuật toán sắp xếp của riêng mình.

Thuật toán của Tí như sau:

```
while (!sorted(a)) {
    int i = random(n);
    int j = random(n);
    if (a[min(i,j)] > a[max(i,j)])
        swap(a[i], a[j]);
}
```

Thuật toán của Tèo:

```
while (!sorted(a)) {
    int i = random(n-1);
    int j = i + 1;
    if (a[i] > a[j])
        swap(a[i], a[j]);
}
```

Cả hai bạn mang thuật toán của mình đến hỏi thầy giáo xem thuật toán của ai tối ưu hơn. Cho trước một dãy số, các bạn hãy tính kỳ vọng của số bước cần thực hiện cho mỗi thuật toán sắp xếp trên để thu được dãy sắp xếp theo thứ tự tăng dần. Tức là với mỗi test, trung bình thuật toán sắp xếp đó sẽ cần bao nhiêu bước thực hiện?

Input

Dòng đầu tiên là số lượng bộ test $2 \leq T \leq 100$.

Mỗi bộ test, đầu tiên là số nguyên dương N ($2 \leq N \leq 8$) là số lượng các phần tử trong dãy.

Tiếp đó là N số nguyên dương nằm trong đoạn $[0; 100]$, các giá trị phần tử trong dãy có thể trùng nhau.

Output

Với mỗi test, in ra trên một dòng giá trị trung bình số bước thực hiện cho thuật toán của Tí và Tèo với độ chính xác 10^{-6} .

Example

Input:

```
12
2 1 2
2 2 1
3 1 2 3
3 3 2 1
4 1 2 3 4
4 4 3 2 1
4 2 1 4 3
5 1 1 1 1 1
5 5 4 3 2 1
```

8 8 7 6 5 4 3 2 1
8 3 1 4 1 5 9 2 6
8 2 7 1 8 2 8 1 8

Output:

Ti 0.000000 Teo 0.000000
Ti 2.000000 Teo 1.000000
Ti 0.000000 Teo 0.000000
Ti 6.000000 Teo 5.000000
Ti 0.000000 Teo 0.000000
Ti 14.666667 Teo 12.500000
Ti 12.000000 Teo 4.500000
Ti 0.000000 Teo 0.000000
Ti 26.382275 Teo 23.641975
Ti 89.576273 Teo 79.496510
Ti 79.161905 Teo 33.422840
Ti 63.815873 Teo 38.910494