

Element with left side smaller and right side greater

给定一个数组arr，找到其中一个元素，该元素左边的数都小于等于该元素，该元素右边的数都大于等于该元素（注：设数组两端元素不满足此条件）

首先设定一个rightMin数组，rightMin[i]表示在i的右边（包括i位置）中最小的元素

然后设定leftMax，从前遍历arr[i]，leftMax表示在i位置的左边（包括i位置）中的最大元素

当 $i \neq N - 1$ 时，且 $arr[i] \geq leftMax$ && $arr[i] \leq rightMin[i]$ 时，返回arr[i]

```
#include <bits/stdc++.h>

using namespace std;

typedef vector<int> vi;

int findEle(vi &arr, int &N) {
    vi rightMin(N);
    rightMin[N - 1] = arr[N - 1];
    for (int i = N - 2; i >= 0; --i) {
        rightMin[i] = min(rightMin[i + 1], arr[i]);
    }

    int leftMax = arr[0];
    for (int i = 1; i < N; ++i) {
        leftMax = max(leftMax, arr[i]);
        if (i != N - 1 && leftMax <= arr[i] && arr[i] <= rightMin[i])
            return arr[i];
    }
    return -1;
}

int main() {
    int T;
    scanf("%d", &T);
    while (T--) {
        int N;
        scanf("%d", &N);
        vi arr(N);
        for (int i = 0; i < N; ++i) scanf("%d", &arr[i]);
        printf("%d\n", findEle(arr, N));
    }
}
```