

```

#include <stdio.h>
#include <iostream>
#include <fstream>
#include <string>

using namespace std;
const unsigned int MAXP=10000;

const int MIN_TEMP=-50;
const int MAX_TEMP=60;

// Pre: 0 <= nTemps <= longitud(temps)
void picosValles(int temps[], int nTemps, int &p, int& v) {

    //INV: p = #i: 0 < i < n: (temps[i - 1] < temps[i]) && (temps[i] > temps[i + 1]) && v
    = #i: 0 <= i < n-1: (temps[i - 1] > temps[i]) && (temps[i] < temps[i + 1])

    p = 0;
    v = 0;

    int n = 1;

    if (nTemps > 2) {

        while (n < nTemps - 1) {

            if ((temps[n - 1] < temps[n]) && (temps[n] > temps[n + 1])) {
                p++;
            }

            if ((temps[n - 1] > temps[n]) && (temps[n] < temps[n + 1])) {
                v++;
            }

            n = n + 1;
        }
    }
}

// Post: p = # i : 0 < i < nTemps-1: (temps[i-1] < temps[i]) && (temps[i] > temps[i+1])
//      && v = # i : 0 < i < nTemps-1: (temps[i-1] > temps[i]) && (temps[i] < temps[i+1])

void resuelveCaso() {

    // Lectura de los datos

    int nTemps;
    cin >> nTemps;

    int temps[MAXP];
    for (int i=0; i<nTemps; i++) {
        cin>>temps[i];
    }

    // Calculo del resultado: una funcion aparte

```

```

int p,v;
picosValles(temps,nTemps,p,v);

// Escritura del resultado

cout << p << " " << v << "\n";
}

int main() {

    // Para la entrada por fichero.
#ifdef DOMJUDGE
    std::ifstream in("casos.txt");
    auto cinbuf = std::cin.rdbuf(in.rdbuf());
#endif

    unsigned int numCasos;
    cin >> numCasos;
    // Resolvemos
    while (numCasos--) {
        resuelveCaso();
    }

#ifdef DOMJUDGE // para dejar todo como estaba al principio
    std::cin.rdbuf(cinbuf);
    system("PAUSE");
#endif

    return 0;
}

```