

```

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

using namespace std;
const unsigned int MAXV = 999;

// Pre: 0 < nNums <= MAXV && nNums = longitud(nums)
pair<long long int, int> encuestas(int nums[], int nNums) {

//Usando minimo(vec, v): min u: 0 <= u < v: vec[u]
//INV: suma = (sumatorio i: 0 <= i < n && nums[i] > minimo(nums, n): nums[i]) && sumat =
(#w: 0 <= w < n: nums[w] > minimo(nums, n))

    long long int suma = nums[0];
    int sumat = 1;
    int n = 1;
    long long int min = nums[0];
    int num = 1;

    while (n < nNums) {

        if (nums[n] == min) {
            num++;
        }
        else if (nums[n] < min) {
            min = nums[n];
            num = 1;
        }

        suma += nums[n];
        sumat++;
        n++;
    }

    suma = suma - (min * num);
    sumat -= num;

    return { suma, sumat };
}

//Usando minimo(vec, j): min u: 0 <= u < j: vec[u]
// Post: suma = (sumatorio i: 0 <= i < nNums && nums[i] > minimo(nums, nNums): nums[i]) &&
sumat = (#w: 0 <= w < nNums: nums[w] > minimo(nums, nNums))

void resuelveCaso() {

    // Lectura de los datos

    int nNums;
    cin >> nNums;

    int nums[MAXV];

```

```

    for (int i = 0; i < nNums; i++)
    {
        cin >> nums[i];
    }

    // Calculo del resultado: una funcion aparte
    pair<long long int, int> sol = encuestas(nums, nNums);

    // Escritura del resultado

    cout << sol.first << " " << sol.second << endl;
}

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;
}

```