

CERINȚĂ:

Fie A un șir de n numere, $n \geq 3$.

Să se scrie un program care să înlocuiască fiecare element nül din șir, a_k , $2 \leq k \leq n-1$, cu:

- a) media geometrică a celui mai mare și a celui mai mic element din șir;
- b) media geometrică a modulului vecinilor săi direcți.

REZOLVARE:

```
#include <cmath>
#include <stdio>
#include <math.h>
#include <vector>
using namespace std;

vector <double> A;

/*
  Fie A un sir de n numere, n >= 3.
  Sa se scrie un program care sa inlocuiasca fiecare element nul din sir Sk, unde 2 <=
  k <= n-1, cu:
*/

// a) Media geomtrica a celui mai mare si a celui mai mic element din sir.
void subpunctA(int n) {

    double elementMinim = DBL_MAX;
    double elementMaxim = -DBL_MAX;

    // Cautam elementul minim al sirului.
    for (int i = 0; i < n; i++)
        if(auxiliar[i] < elementMinim)
            elementMinim = auxiliar[i];

    // Cautam elementul maxim al sirului.
    for (int i = 0; i < n; i++)
        if(auxiliar[i] > elementMaxim)
            elementMaxim = auxiliar[i];

    // Verificam daca putem aplica regula pe sirul dat.
    // In caz negativ, opriam algoritmul si afisam un mesaj corespunzator.
    if(elementMinim * elementMaxim < 0) {
        printf("Nu putem rezolva subpunctul a) deoarece produsul celui mai mic numar si al
        celui mai mare numar este negativ!\n");
        return;
    }
    else {
        // Afisare sir obtinut.
```

```

    printf("Vectorul obtinut in urma subpunctul a): ");
    for (int i = 0; i < n; i++)
        printf("%.2lf ", A[i]);
    printf("\n");
}

// b) Media geometrica a modulului vecinilor sai directi.
void subpunctB(int n) {

    vector <double> auxiliar;
    auxiliar = A;
    double valoareDeInlocuit;

    // Cautam valorile nule pentru a le inlocui in mod corespunzator.
    for (int i = 1; i < n-1; i++) {
        if(auxiliar[i] == 0) {
            valoareDeInlocuit = auxiliar[i-1] * auxiliar[i + 1];
            // Verificam daca putem aplica regula pe sirului dat.
            // In caz negativ, opresc algoritmul si afisam un mesaj corespunzator.
            if (valoareDeInlocuit < 0) {
                valoareDeInlocuit = valoareDeInlocuit * (-1);
            }
            auxiliar[i] = sqrt(valoareDeInlocuit);
        }
    }

    // Afisare sir obtinut.
    printf("Vectorul obtinut in urma subpunctul b): ");
    for (int i = 0; i < n; i++)
        printf("%.2lf ", auxiliar[i]);
    printf("\n");
}

int main() {

    freopen("date.in", "r", stdin);
    freopen("date.out", "w", stdout);
    int n;
    double x;

    // Citirea datelor de intrare.
    scanf("%d", &n);

    for (int i = 0; i < n; i++) {
        scanf("%lf", &x);
        A.push_back(x);
    }

    // Apelarea rezolvarilor pentru cele doua subpuncte.
    subpunctA(n);
    subpunctB(n);

    return 0;
}

```

EFICIENȚĂ ȘI ANALIZĂ COMPLEXITATE:

Problema dată poate fi abordată în mod direct.

În primul rând citim datele de intrare, iar de aici ne reies $n + 1$ pași.

Pentru primul subpunct începem să parcurgem vectorul citit în căutarea minimului și a maximului, fapt din care rezulta maxim $2*n$ pași în total. Ulterior, verificăm dacă produsul dintre elementul minim și cel maxim este negativ. În caz afirmativ, nu avem decât să returnăm vectorul inițial astfel încât vom mai executa n pași. În caz negativ, vom returna un mesaj de avertisment întrucât nu putem face radical dintr-o valoare negativă.

În final, pentru primul subpunct putem avea maxim $3*n$ pași.

Pentru cel de-al doilea subpunct parcurgem vectorul, iar în momentul în care întâlnim o valoare nulă încercăm să calculăm media geometrică a modulului vecinilor săi direcți. Astfel, vom avea de efectuat maxim $n + (n-2)$ pași, deoarece parcurgem toate elementele și putem avea maxim $n - 2$ valori nule.

În final, pentru cel de-al doilea subpunct putem avea maxim $2 * n - 2$ pași.

Așadar, vom avea $n + 1 + 3*n + 2 * n - 2 = 6 * n - 1$ pași, complexitate liniară ce poate fi redusă la $O(n)$.

Știm că această soluție este optimă întrucât avem nevoie de minim n pași (pentru citirea datelor), deci nu putem ajunge la ceva mai eficient.

EXPLICAȚII:

Subpunctul a)

Pentru primul subpunct, distingem mai multe cazuri:

- dacă minimul este negativ și maximul pozitiv, atunci produsul celor două este negativ deci nu le putem face media geometrică
- dacă minimul este negativ și maximul este 0, atunci toate valorile de zero vor fi înlocuite tot cu zero, deci vectorul este neschimbat
- dacă minimul este zero și maximul este pozitiv, atunci valorile de zero vor fi înlocuite tot cu zero, deci vectorul este neschimbat
- dacă minimul este negativ și maximul este negativ înseamnă că nu avem niciun zero, deci nu trebuie să înlocuim nimic și vectorul este neschimbat
- dacă minimul este pozitiv și maximul este pozitiv înseamnă că nu avem niciun zero, deci nu trebuie să înlocuim nimic și vectorul este neschimbat
- dacă atât minimul cât și maximul sunt 0 înseamnă că avem vectorul constant zero, deci vectorul rămâne neschimbat

Subpunctul b)

Pentru cel de-al doilea subpunct vom putea duce întotdeauna soluția la capăt întrucât avem de făcut media geometrică a valorii în modul a produsului celor doi vecini direcți ai fiecărui element.