

# PROIECT DE PROGRAME NR. 3. ORGANIZAREA CICLURILOR ÎN PROGRAM. LUCRUL CUTABELE DE NUMERE. POINTERI. FUNCȚII

## I. Formularea problemei

De efectuat următoarele operații conform variantei:

1. De determinat într-un tabel de numere valorile minimă și maximă.
2. De determinat într-un tabel de numere media aritmetică.
3. De aranjat un tabel de numere în ordine crescătoare sau descrescătoare, utilizând una dintre metodele de sortare.
4. De efectuat rotirea la stânga (variantele impare) sau la dreapta (variantele pare) a elementelor tabelului de k ori ( $k$  – numărul variantei).
5. De adăugat date în tabelul de numerele.
6. De șters date din tabelul de numerele.
7. De calculat o caracteristică, referitoare la datele concrete (se precizează individual).

## II. Indicații metodice

1. De scris funcții pentru citirea, afișarea și prelucrarea tabelelor statice, citite de la tastatură sau generate aleator.
2. Programul trebuie să fie universal, adică trebuie să se obțină rezultatul pentru orice date inițiale.
3. De utilizat pointeri. De scris funcții, care returnează pointeri.

## III. Programul

```
#include <iostream>
#include <cstdlib>
#include <conio.h>
#include <cmath>

using namespace std;

int rrand(int range_min, int range_max) {
    return rand() % (range_max - range_min + 1) + range_min;
}

int WriteArr(int arr[], int size) {
    for (int i = 0; i < size; ++i) {
        cout << arr[i] << " ";
    }
    return 0;
}

int SortArray(int arr[], int size, int ord) {
```

```

int temp;
for (int i = 0; i < size - 1; i++) {
    for (int j = 0; j < size - i - 1; j++) {
        if (ord == 1) {
            if (arr[j] > arr[j + 1]) {
                temp = arr[j];
                arr[j] = arr[j + 1];
                arr[j + 1] = temp;
            }
        } else {
            if (arr[j] < arr[j + 1]) {
                temp = arr[j];
                arr[j] = arr[j + 1];
                arr[j + 1] = temp;
            }
        }
    }
}
WriteArr(arr, size);
return 0;
}

int DeleteElement(int x[], int k, int &n) {
    for (int i = k; i < n; ++i) {
        x[i] = x[i + 1];
    }
    n--;

    WriteArr(x, n);

    return 0;
}

int AddElement(int x[], int h, int &n) {
    n++;

    x[n - 1] = h;

    WriteArr(x, n);
    return 0;
}

int Rotire(int x[], int z, int n) {
    z = x[0];
    for (int i = 1; i <= n; i++) x[i - 1] = x[i];
    x[n - 1] = z;
    cout << "Vectorul obtinut" << "\n";
    WriteArr(x, n);
    return 0;
}

float MediaNotelor(int arr[], int size) {
    int *n = &size;
    float *medie = new float;
    int *arr_ptr = arr;

    for (int i = 0; i < size; ++i) {
        *medie += *(arr_ptr + i);
    }

    *medie = *medie / *n;

    return *medie;
}

```

```

}

int MinMax(int arr[], int size, int item) {
    int *min = new int;
    int *max = new int;
    int *arr_ptr = arr;

    *min = arr[0];
    *max = arr[0];

    for (int i = 0; i < size; ++i) {
        if (*(arr_ptr + i) > *max) {
            *max = *(arr_ptr + i);
        }
        if (*(arr_ptr + i) < *min) {
            *min = *(arr_ptr + i);
        }
    }

    if (item == 1) {
        return *max;
    } else {
        return *min;
    }
}

float Calitate(int x[], int n) {
    float calitatie = 0;
    for (int i = 0; i < n; ++i) {
        if (x[i] == 10 || x[i] == 9)
            calitatie++;
    }
    return round((calitatie / n * 100) * 100) / 100;
}

int main() {
    int z, k, h, n, n2, ord;
    cout << "\n" << "Introduceti nr de elevi" << "\n";
    cin >> n;

    int x[n];

    for (int i = 0; i < n; ++i) {
        x[i] = rand(1, 10);
    }

    cout << "\n" << "Vectorul introdus" << WriteArr(x, n) << "\n";

    cout << "\n" << "Introduceti ordinea sortarii 1 sau -1 " << "\n";
    cin >> ord;
    SortArray(x, n, ord);

    cout << "\n" << "Media notelor " << MediaNotelor(x, n) << "\n";

    cout << "\n" << "Introduceti 1 pentru a afla numarul maxim si 0 pentru
    minim" << "\n";
    cin >> n2;
    if (n2 == 1) {
        cout << "\n" << "Numarul maxim este " << MinMax(x, n, n2) << "\n";
    } else {
        cout << "\n" << "Numarul minim este " << MinMax(x, n, n2) << "\n";
    }
}

```

```

cout << "\n" << "Introduceti nr elementului pentru stergere" << "\n";
cin >> k;
DeleteElement(x, k, n);

cout << "Introduceti noul element" << endl;
cin >> h;
AddElement(x, h, n);

cout << "Rotirea vectorului cu o pozitie la stanga" << "\n";
Rotire(x, z, n);

cout << "%Calitatii: " << Calitate(x, n) << "%" << endl;

return 0;
}

```

#### IV. Rezultate

Introduceti nr de elevi

17

Vectorul introdus

2 8 5 1 10 5 9 9 3 5 6 6 2 8 2 2 6

Introduceti ordinea sortarii 1 sau -1

1

Vectorul obtinut

1 2 2 2 2 3 5 5 5 6 6 6 8 8 9 9 10

Media notelor 5.23529

Introduceti 1 pentru a afla numarul maxim si 0 pentru minim

1

Numarul maxim este 10

Introduceti nr elementului pentru stergere

1

Vectorul obtinut

2 2 2 2 3 5 5 5 6 6 6 8 8 9 9 10

Introduceti noul element

4

Vectorul obtinut

2 2 2 2 3 5 5 5 6 6 6 8 8 9 9 10 4

Rotirea vectorului cu o pozitie la stanga

Vectorul obtinut

2 2 2 3 5 5 5 6 6 6 8 8 9 9 10 4 2

%Calitatii: 17.65%