

Лабораторна робота №4

Тема: Реалізація внутрішнього сортування масивів. Метод бульбашки, вибору та вставки.

Мета роботи: вивчення масивів, як структури організації даних, набуття навичок їх створення та опрацювання.

Хід роботи

Завдання для виконання

1. Написати програму сортування масиву методом бульбашки.

Лістинг 4.1 – Код програми

```
// Чавурська Єлизавета, П-31, АiСД, Варіант 20, Лр 4
#include <iostream>
using namespace std;
int main() {
    setlocale(LC_CTYPE, "Ukr");
    int arr[] = { 5, 2, 9, 1, 5, 6 };
    int n = sizeof(arr) / sizeof(arr[0]);
    cout << "Початковий масив: ";
    for (int i = 0; i < n; i++) {
        cout << arr[i] << " ";
    }
    cout << endl;
    // Алгоритм бульбашкового сортування
    for (int i = 0; i < n - 1; i++) {
        for (int j = 0; j < n - i - 1; j++) {
            if (arr[j] > arr[j + 1]) {
                // обмін елементів
                int temp = arr[j];
                arr[j] = arr[j + 1];
                arr[j + 1] = temp;
            }
        }
    }
    cout << "Відсортований масив: ";
    for (int i = 0; i < n; i++) {
        cout << arr[i] << " ";
    }
    cout << endl;
    cout << "Чавурська Єлизавета, П-31, АiСД, Варіант 20, Лр 4";
    return 0;
}
```

Початковий масив: 5 2 9 1 5 6
В?дсортований масив: 1 2 5 5 6 9
Чавурська Єлизавета, П-31, А?СД, Вар?ант 20, Лр 4

Рисунок 4.1 – Вікно результатів програми

2. Написати програму сортування масиву методом вибору.

Лістинг 4.2 – Код програми

```
// Чавурська Єлизавета, П-31, АiСД, Варіант 20, Лр 4
#include <iostream>
using namespace std;
int main() {
    setlocale(LC_CTYPE, "Ukr");
    int arr[] = { 64, 25, 12, 22, 11 };
    int n = sizeof(arr) / sizeof(arr[0]);
    cout << "Початковий масив: ";
    for (int i = 0; i < n; i++) {
        cout << arr[i] << " ";
    }
    cout << endl;
    // Алгоритм сортування методом вибору
    for (int i = 0; i < n - 1; i++) {
        int minIndex = i;
        // Пошук найменшого елемента у невідсортованій частині
        for (int j = i + 1; j < n; j++) {
            if (arr[j] < arr[minIndex]) {
                minIndex = j;
            }
        }
        int temp = arr[i];
        arr[i] = arr[minIndex];
        arr[minIndex] = temp;
    }
    cout << "Відсортований масив: ";
    for (int i = 0; i < n; i++) {
        cout << arr[i] << " ";
    }
    cout << endl;
    cout << "Чавурська Єлизавета, П-31, АiСД, Варіант 20, Лр 4";
    return 0;
}
```

Початковий масив: 64 25 12 22 11
В?дсортований масив: 11 12 22 25 64
Чавурська Єлизавета, П-31, А?СД, Вар?ант 20, Лр 4

Рисунок 4.2 – Вікно результатів програми

3. Написати програму сортування масиву методом вставки.

Лістинг 4.3 – Код програми

```
// Чавурська Єлизавета, П-31, АiСД, Варіант 20, Лр 4
#include <iostream>
using namespace std;
int main() {
    setlocale(LC_CTYPE, "Ukr");
    int arr[] = { 12, 11, 13, 5, 6 };
    int n = sizeof(arr) / sizeof(arr[0]);
    cout << "Початковий масив: ";
    for (int i = 0; i < n; i++) {
        cout << arr[i] << " ";
    }
    cout << endl;
    // Алгоритм сортування методом вставки
    for (int i = 1; i < n; i++) {
        int key = arr[i];
        int j = i - 1;
        while (j >= 0 && arr[j] > key) {
            arr[j + 1] = arr[j];
            j--;
        }
        arr[j + 1] = key;
    }
    cout << "Відсортований масив: ";
    for (int i = 0; i < n; i++) {
        cout << arr[i] << " ";
    }
    cout << endl;
    cout << "Чавурська Єлизавета, П-31, АiСД, Варіант 20, Лр 4";
    return 0;
}
```

Початковий масив: 12 11 13 5 6
 Відсортований масив: 5 6 11 12 13
 Чавурська Єлизавета, П-31, АiСД, Варіант 20, Лр 4

Рисунок 4.3 – Вікно результатів програми

4. Протестувати кожну програму на не менш як чотирьох різних вхідних даних:
- вхідний масив заповнено випадковим чином;
 - вхідний масив має відсортовану ліву частину;
 - вхідний масив має відсортовану праву частину;
 - вхідний масив відсортовано у протилежному порядку тощо.

Лістинг 4.4 – Код програми

```
// Чавурська Єлизавета, П-31, АиСД, Варіант 20, Лр 4
#include <iostream>
using namespace std;

// Функція сортування методом вставки
void func(int arr[], int n) {
    for (int i = 1; i < n; i++) {
        int key = arr[i];
        int j = i - 1;
        while (j >= 0 && arr[j] > key) {
            arr[j + 1] = arr[j];
            j--;
        }
        arr[j + 1] = key;
    }
}

// Функція для виводу масиву
void printArray(int arr[], int n) {
    setlocale(LC_CTYPE, "Ukr");
    for (int i = 0; i < n; i++) cout << arr[i] << " ";
    cout << endl;
}

int main() {
    // 1 Випадковий масив
    int arr1[] = { 7, 2, 9, 1, 5 };
    int n1 = sizeof(arr1) / sizeof(arr1[0]);
    cout << "1. Випадковий масив: ";
    printArray(arr1, n1);
    func(arr1, n1);
    cout << " Відсортовано: ";
    printArray(arr1, n1);
    cout << endl;

    // 2 Відсортована ліва частина
    int arr2[] = { 1, 3, 5, 8, 7, 2, 4 };
    int n2 = sizeof(arr2) / sizeof(arr2[0]);
    cout << "2. Відсортована ліва частина: ";
    printArray(arr2, n2);
    func(arr2, n2);
    cout << " Відсортовано: ";
    printArray(arr2, n2);
    cout << endl;

    // 3 Відсортована права частина
    int arr3[] = { 8, 4, 2, 1, 5, 7, 9, 10 };
    int n3 = sizeof(arr3) / sizeof(arr3[0]);
    cout << "3. Відсортована права частина: ";
    printArray(arr3, n3);
    func(arr3, n3);
    cout << " Відсортовано: ";
    printArray(arr3, n3);
}
```

```

cout << endl;

// 4 Масив у протилежному порядку
int arr4[] = { 10, 9, 8, 7, 6, 5, 4 };
int n4 = sizeof(arr4) / sizeof(arr4[0]);
cout << "4. Масив у протилежному порядку: ";
printArray(arr4, n4);
func(arr4, n4);
cout << " Відсортовано: ";
printArray(arr4, n4);
cout << endl;
cout << "Чавурська Єлизавета, П-31, АiСД, Варіант 20, Лр 4";
return 0;
}

```

1. Тшярфьютш ѿрёшт: 7 2 9 1 5
В?дсортовано: 1 2 5 7 9
2. В?дсортована л?ва частина: 1 3 5 8 7 2 4
В?дсортовано: 1 2 3 4 5 7 8
3. В?дсортована права частина: 8 4 2 1 5 7 9 10
В?дсортовано: 1 2 4 5 7 8 9 10
4. Масив у протилежному порядку: 10 9 8 7 6 5 4
В?дсортовано: 4 5 6 7 8 9 10

Чавурська Єлизавета, П-31, А?СД, Вар?ант 20, Лр 4

Рисунок 4.4 – Вікно результатів програми

5. Дописати кілька рядків коду до останньої програми для визначення характеристики масиву за варіантом з Додатку.

Додаток

Варіант	Вигляд масиву	Характеристика масиву	Блок – схема методу
20	Масив з 30 дійсних чисел в інтервалі (-20..20)	Знайти кількість елементів у інтервалі від -13 до 13	Вибору

Лістинг 4.5 – Код програми

```

// Чавурська Єлизавета, П-31, АiСД, Варіант 20, Лр 4
#include <iostream>
#include <iomanip>
using namespace std;
int main() {

```

```
setlocale(LC_CTYPE, "Ukr");
const int N = 30;
double arr[N] = {
    -15.3, 12.7, 8.1, -4.6, 19.9, -20.0, 5.5, -11.2, 2.4, 17.3,
    -3.5, 14.6, -8.8, 9.2, -19.4, 6.7, 0.0, 11.1, -12.3, 13.5,
    -9.9, 4.3, -7.2, 10.4, -18.6, 3.6, 7.5, -1.9, 15.0, -14.8
};
cout << fixed << setprecision(2);
cout << "Початковий масив:\n";
for (int i = 0; i < N; i++) {
    cout << setw(7) << arr[i];
    if ((i + 1) % 10 == 0) cout << endl;
}
// 1 Сортування методом вибору
for (int i = 0; i < N - 1; i++) {
    int minIndex = i;
    for (int j = i + 1; j < N; j++) {
        if (arr[j] < arr[minIndex])
            minIndex = j;
    }
    double temp = arr[i];
    arr[i] = arr[minIndex];
    arr[minIndex] = temp;
}
cout << "\nВідсортований масив:\n";
for (int i = 0; i < N; i++) {
    cout << setw(7) << arr[i];
    if ((i + 1) % 10 == 0) cout << endl;
}
// 2 Підрахунок елементів у діапазоні [-13; 13]
int count = 0;
for (int i = 0; i < N; i++) {
    if (arr[i] >= -13 && arr[i] <= 13)
        count++;
}

cout << "\nКількість елементів у інтервалі [-13; 13]: " << count << endl;
cout << "Чавурська Єлизавета, П-31, АiСД, Варіант 20, Лр 4";
return 0;
}
```

Початковий масив:

-15.30	12.70	8.10	-4.60	19.90	-20.00	5.50	-11.20	2.40	17.30
-3.50	14.60	-8.80	9.20	-19.40	6.70	0.00	11.10	-12.30	13.50
-9.90	4.30	-7.20	10.40	-18.60	3.60	7.50	-1.90	15.00	-14.80

В?дсортований масив:

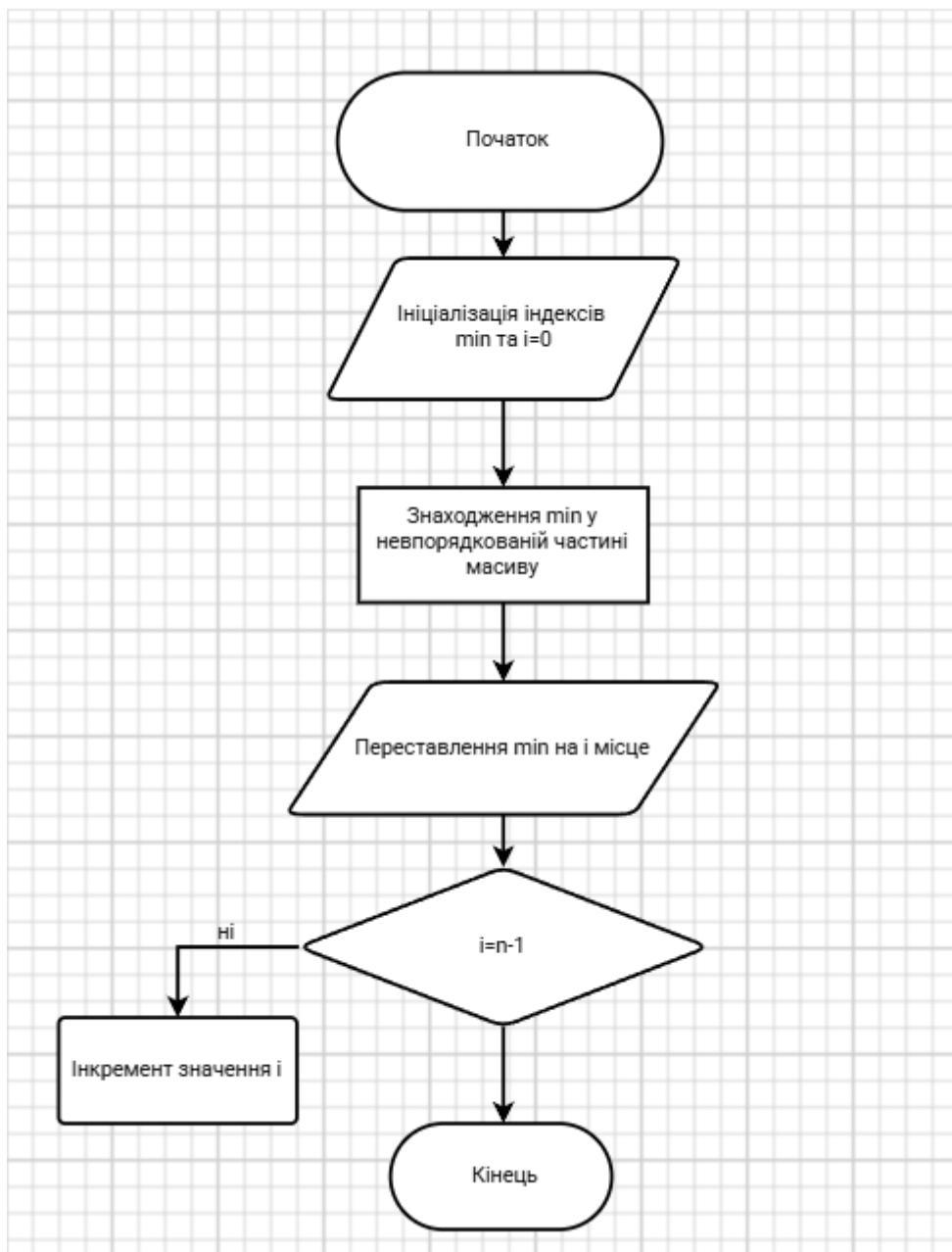
-20.00	-19.40	-18.60	-15.30	-14.80	-12.30	-11.20	-9.90	-8.80	-7.20
-4.60	-3.50	-1.90	0.00	2.40	3.60	4.30	5.50	6.70	7.50
8.10	9.20	10.40	11.10	12.70	13.50	14.60	15.00	17.30	19.90

К?льк?сть елемент?в у ?нтервал? [-13; 13]: 20

Чавурська Єлизавета, П-31, А?СД, Вар?ант 20, Пр 4

Рисунок 4.5 – Вікно результатів програми

6. Записати нижню, верхню та асимптотичну оцінку складності кожного методу.
7. Намалювати блок – схему сортування одновимірного масиву одним із методів за варіантом з Додатку.



Блок-схема 4.1

8. Зробити висновок про роботу та записати у звіт.

Висновок: на цій лабораторній роботі я вивчила масиви, як структури організації даних, набула навичок їх створення та опрацювання.