МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

ФАКУЛЬТЕТ ІНФОРМАТИКИ ТА ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ

КАФЕДРА ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ

Лабораторна робота №5

з дисципліни «Алгоритми та структури даних»

Виконав:

студент групи ІМ-o51

Машута Олександр Ігорович

Перевірив:

ас. Череватенко О.В.

Київ 2025

**Мета лабораторної роботи**

Метою лабораторної роботи №5 є засвоєння теоретичного матеріалу та набуття практичних навичок рішення задачі сортуван - ня заданої категорії елементів за допомогою різних алгоритмів сортування у двовимірних масивах.

**Постановка задачі**

1. Задано двовимірний масив (матрицю) цілих чисел ***A***[***m***,***n***] або ***A***[***n***,***n***], де ***m*** та ***n*** – натуральні числа (константи), що визначають розміри двовимірного масиву. Виконати сортування цього масиву або заданої за варіантом його частини у заданому порядку заданим алгоритмом (методом).

***Сортування повинно бути виконано безпосередньо у двовимірному масиві «на тому ж місці»***, тобто без перезаписува- ння масиву та/або його будь-якої частини до інших одно- або двовимірних масивів, а також без використання спискових структур даних.

1. Розміри матриці ***m*** та ***n*** взяти самостійно у межах від 7 до 10.

При тестуванні програми необхідно підбирати такі вхідні набори початкових значеннь матриці, щоб можна було легко від- стежити коректність виконання сортування і ця коректність була б протестована для всіх можливих випадків. З метою тестування до- зволяється використовувати матриці меншого розміру.

Текст програми:

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

#define N 7 // Розмір квадратної матриці(може бути змінений)

int matrix[N][N];

// Функція генерації випадкових чисел у заданому діапазоні

int RandomRange(const int min, const int max) {

return rand() % (max - min + 1) + min;

}

// Заповнення матриці випадковими значеннями

void FillRandomMatrix() {

for (int i = 0; i < N; i++) {

for (int j = 0; j < N; j++) {

matrix[i][j] = RandomRange(-100, 100);

}

}

}

// Вивід матриці на екран

void PrintMatrix() {

printf("\n");

for (int i = 0; i < N; i++) {

for (int j = 0; j < N; j++) {

printf("%5d", matrix[i][j]);

}

printf("\n");

}

printf("\n");

}

// Сортування головної діагоналі методом Шелла

void ShellSortMainDiagonal() {

int temp;

for (int gap = N / 2; gap > 0; gap /= 2) {

for (int i = gap; i < N; i++) {

temp = matrix[i][i];

int j;

for (j = i; j >= gap && matrix[j - gap][j - gap] > temp; j -= gap) {

matrix[j][j] = matrix[j - gap][j - gap];

}

matrix[j][j] = temp;

}

}

}

int main(void) {

srand((unsigned int)time(NULL));

FillRandomMatrix();

printf("Початкова матриця:");

PrintMatrix();

ShellSortMainDiagonal();

printf("Матриця після сортування головної діагоналі:");

PrintMatrix();

return 0;

}

Тести програми:

A screenshot of a computer

AI-generated content may be incorrect.

A table of numbers with black text

AI-generated content may be incorrect.

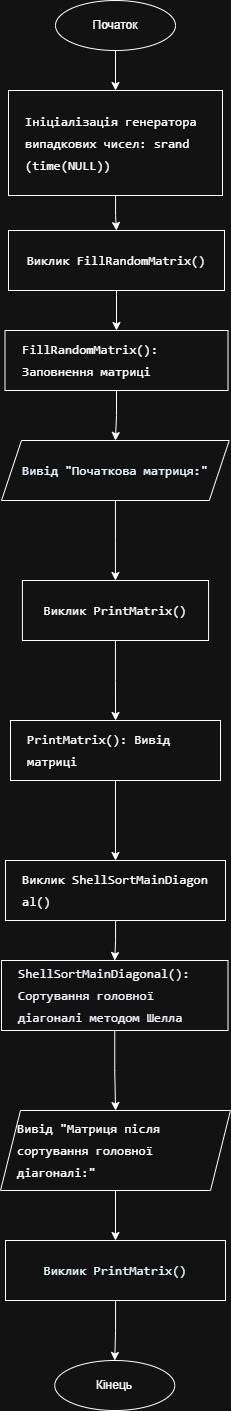
A screenshot of a math game

AI-generated content may be incorrect.

A screenshot of a math game

AI-generated content may be incorrect.

Блок-схема



**Висновок**

Під час виконання лабораторної роботи №6 я навчився сортувати головну діагональ квадратної матриці цілих чисел за допомогою методу Шелла.

Я дізнався, що метод Шелла — це ефективний спосіб сортування, який працює швидше за звичайне сортування вставками, завдяки тому, що порівнює та переміщує елементи на більші відстані.

В ході роботи я написав програму, яка сортує лише головну діагональ матриці за зростанням. Це допомогло мені краще зрозуміти, як працюють алгоритми сортування та як їх застосовувати до двовимірних масивів.

Таким чином, ця лабораторна робота була корисною і допомогла мені отримати практичні навички програмування мовою C.