ДНІПРОВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ОЛЕСЯ ГОНЧАРА

ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

КАФЕДРА КОМП’ЮТЕРНИХ ТЕХНОЛОГІЙ

**Лабораторна робота №8**

«Двовимірні масиви. Робота з файлами»

**з курсу «Програмування»**

**Варіант №14**

Виконав:

студент групи ПА-22-2

Овдієнко Андрій Володимирович

Дніпро, 2023

**Зміст**

[**Зміст** 2](#_Toc133081242)

[**1.Завдання до лабораторної роботи №8** 3](#_Toc133081243)

[**2.Опис розв’язку** 4](#_Toc133081244)

[**Задача 2.1** 4](#_Toc133081245)

[**Задача 2.2** 5](#_Toc133081246)

[**Задача 2.3** 6](#_Toc133081247)

[**3.Опис інтерфейсу програми (керівництво користувача)** 9](#_Toc133081248)

[**Приклад заповнення з консолю (без запису у файл / із записом у файл)** 9](#_Toc133081249)

[**Приклад 1.1 (без запису у файл):** 9](#_Toc133081250)

[**Приклад 1.2 (із записом у файл “lab8\_1.txt”):** 11](#_Toc133081251)

[**Приклади заповнення з файлу** 14](#_Toc133081252)

[**Приклад 2.1 (без запису у файл)** 17](#_Toc133081253)

[**Приклад 2.2 (із записом у файл “lab8\_2\_2.txt”)** 19](#_Toc133081254)

[**Додаток** 22](#_Toc133081255)

[**Код програми** 22](#_Toc133081256)

**1.Завдання до лабораторної роботи №8**

**Завдання 1**

Складіть програму, яка здійснює обробку вхідних даних за допомогою двовимірних масивів. При виконанні лабораторної роботи слід врахувати наступне:

1. Дати користувачеві можливість вибору: вводити вхідні дані з консолі, чи вводити їх з файлу; у випадку вводу з файлу продемонструвати дві можливості: а) можливість користувача задати ім’я файлу в діалозі; б) можливість перенапрямлення потоку вводу/виводу на файл в командному рядку.
2. При програмуванні однотипних підзадач (сортування, ввід/вивід масивів, знаходження суми стовпця матриці тощо) слід оформлювати їх у вигляді функцій.

**Завдання 2**

**Варіант 14.** Задайте матриці *А* і *В* розмірністю (*n* x *m*), де *n* і *m* ≥ 6. Виконайте такі завдання:

1. знайдіть мінімальний елемент матриці А і замініть ним всі від’ємні елементи матриці B;
2. знайдіть вектор С, i-й елемент якого являє собою суму елементів i-го рядка матриці А;
3. упорядкуйте за зростанням значення елементів стовпця, де знаходиться мінімальний елемент матриці А.

**2.Опис розв’язку**

**Задача 2.1**

Надаємо змінній “min” перший елемент масиву, який розташований у першому рядку та в першій колонці, але так-як у масивів індекси починаються не з 1, а з 0, то ми надаємо значення нульового рядка та нульового стовпця масиву А змінній “min”:

min = A[0][0];

Обходимо всі значення (рядки та стовпці) масиву (матриці) А[n1][m1], де n1 і m1 – кількість рядків і стовпців у масиві:

for (int i=0; i<n1; i++)

{

for (int j=0; j<m1; j++)

{

}

}

Під час обходу всіх значень елементу масиву А, порівнюємо мінімальне значення (min) з кожним елементом матриці А[n1][m1], де n1 і m1 – кількість рядків і стовпців у масиві, якщо мнімальне значення елементу більше за значення масиву A, то “min” (яке спочатку приймає перше значення масиву - A[0][0]) приймає значення того елементу за який він більший, таким чином знайдемо мінімальний елемент матриці А:

if (A[i][j]< min)

{

min = A[i][j];

}

Тепер обходимо всі значення (рядки та стовпці) масиву (матриці) B[n2][m2], де n2 і m2 – кількість рядків і стовпців у масиві:

for (int i=0; i<n2; i++)

{

for (int j=0; j<m2; j++)

{

}

}

Порівнюєм кожний елемент масиву B з нулем, щоб визначити, який з елементів є відʼємним, якщо елемент масиву B строго менший за нуль, то замінюємо елемент ції матриці на найменше значення масиву А (“min”):

if (B[i][j]< 0)

{

B[i][j] = min;

}

Якщо скомпонувати розвʼязок задачі 2.1, де потрібно знайти мінімальний елемент матриці А і замінити ним всі від’ємні елементи матриці B, то вийде наступне:

min = A[0][0];

for (int i=0; i<n1; i++)

{

for (int j=0; j<m1; j++)

{

if (A[i][j]< min)

{

min = A[i][j];

}

}

}

for (int i=0; i<n2; i++)

{

for (int j=0; j<m2; j++)

{

if (B[i][j]< 0)

{

B[i][j] = min;

}

}

}

**Задача 2.2**

Обходимо всі значення (рядки та стовпці) масиву (матриці) А[n1][m1], де n1 і m1 – кількість рядків і стовпців у масиві, а також вводимо змінну “sum” яка приймає значення 0 на початку кожного рядку:

for (int i=0; i<n1; i++)

{

sum = 0;

for (int j=0; j<m1; j++)

{

}

}

У двох циклах знаходимо вектор С, i-й елемент якого являє собою суму елементів i-го рядка матриці А (складаємо значення “sum” та значень матриці А[n1][m1] у значення “sum”) а в кінці рядку надаємо певному значенню вектора С[n1] значення “sum”:

for (int j=0; j<m1; j++)

{

sum += A[i][j];

}

C[i] = sum;

Якщо скомпонувати розвʼязок задачі 2.2, де потрібно знайти вектор С, i-й елемент якого являє собою суму елементів i-го рядка матриці А, то вийде наступне:

for (int i=0; i<n1; i++)

{

int sum = 0;

for (int j=0; j<m1; j++)

{

sum += A[i][j];

}

C[i] = sum;

}

**Задача 2.3**

Обходимо всі значення (рядки та стовпці) масиву (матриці) А[n1][m1], де n1 і m1 – кількість рядків і стовпців у масиві так, щоб не вийти за межі матриці:

for (int i =0; i<n1-1; i++)

{

for(int j=i+1; j<n1; j++)

{

}

}

Візьмемо код з задачі 2.1, і під час знахлдження мінімального елемента матриці A[n1][m1], знайдемо стовпець у якому він розташований за допомогою змінної “o2” прирівнявши її до змінної “j” (індекса стовпця матриці):

min = A[0][0];

for (int i=0; i<n1; i++)

{

for (int j=0; j<m1; j++)

{

if (A[i][j]< min)

{

min = A[i][j];

o2 = j;

}

}

}

Перевіряємо значення стовпцю масиву A, де знахлдиться мінімальний елемент. Якщо значення елементу більший за наступний елемент у цьому стовпці, то змінюємо рядки матриці місцями:

if (A[i][o2]>A[j][o2])

{

for (int k=0; k<m1; k++)

{

swap(A[i][k], A[j][k]);

}

}

Якщо скомпонувати розвʼязок задачі 2.3, де потрібно упорядкувати за зростанням значення елементів стовпця, де знаходиться мінімальний елемент матриці А, то вийде наступне:

for (int i =0; i<n1-1; i++)

{

for(int j=i+1; j<n1; j++)

{

if (A[i][o2]>A[j][o2])

{

for (int k=0; k<m1; k++)

{

swap(A[i][k], A[j][k]);

}

}

}

}

**3.Опис інтерфейсу програми (керівництво користувача)**

**Приклад заповнення з консолю (без запису у файл / із записом у файл)**

**Приклад 1.1 (без запису у файл):**

Mатриці А та В розміром n x m, де n і m ≥ 6.

Розмірність матриці А[n][m]:

n = 6

m = 7

Розмірність матриці B[n][m]:

n = 7

m = 8

Заповнити матрицю із файлу? (y/n) n

Матриця A[6][7]:

| 67 64 -17 78 88 -73 82 |

| 55 -86 77 9 46 22 -91 |

| 26 67 -66 81 35 46 -11 |

| -53 12 -48 -59 -48 34 82 |

| -42 -28 -90 69 -2 82 82 |

| -53 -61 -21 29 -39 -100 22 |

Матриця B[7][8]:

| -86 30 -31 -3 78 28 47 -74 |

| 65 -4 72 -10 10 77 0 15 |

| -44 1 95 32 67 89 52 86 |

| -29 38 -2 -5 18 -51 -97 -65 |

| 54 27 -41 69 -72 -52 -93 23 |

| -80 89 -78 58 -9 -84 -57 46 |

| 89 52 -20 73 -69 -81 -31 51 |

Мінімальний елемент матриці А[6][8] = -100, де цей елемент в матриці А[6][6].

Mінімальний елементом матриці А замінив всі від’ємні елементи матриці B.

Матриця B[7][8]:

| -100 30 -100 -100 78 28 47 -100 |

| 65 -100 72 -100 10 77 0 15 |

| -100 1 95 32 67 89 52 86 |

| -100 38 -100 -100 18 -100 -100 -100 |

| 54 27 -100 69 -100 -100 -100 23 |

| -100 89 -100 58 -100 -100 -100 46 |

| 89 52 -100 73 -100 -100 -100 51 |

Кожний i-й елемент лінійного вектора С являє собою суму елементів i-го рядка матриці А.

С = | 289 32 178 -80 71 -223 |

Мінімальне значення матриці А[6][7] = -100, де цей елемент в матриці А[6][6].

Матриця А[6][7] після упорядкування за зростанням значення елементів стовпця, де знаходиться мінімальний елемент матриці:

Матриця A[6][7]:

| -53 -61 -21 29 -39 -100 22 |

| 67 64 -17 78 88 -73 82 |

| 55 -86 77 9 46 22 -91 |

| -53 12 -48 -59 -48 34 82 |

| 26 67 -66 81 35 46 -11 |

| -42 -28 -90 69 -2 82 82 |

Записати дані у файл? (y/n) n

**Приклад 1.2 (із записом у файл “lab8\_1.txt”):**

Mатриці А та В розміром n x m, де n і m ≥ 6.

Розмірність матриці А[n][m]:

n = 6

m = 6

Розмірність матриці B[n][m]:

n = 7

m = 7

Заповнити матрицю із файлу? (y/n) n

Матриця A[6][6]:

| 74 -15 -81 43 -12 -37 |

| -80 -2 60 -81 90 -83 |

| 97 68 -13 47 53 -79 |

| -29 72 -29 -26 95 6 |

| 29 57 -69 38 -68 62 |

| 8 76 30 62 54 -88 |

Матриця B[7][7]:

| -59 -51 -6 -27 -19 -8 -12 |

| -13 48 -49 79 -54 39 64 |

| 31 34 -67 -55 0 -29 -66 |

| 82 3 -58 -96 10 -88 -30 |

| -13 23 10 -99 58 -32 54 |

| 68 55 22 -2 48 -12 -93 |

| 6 54 88 -77 -12 -85 -18 |

Мінімальний елемент матриці А[6][7] = -88, де цей елемент в матриці А[6][6].

Mінімальний елементом матриці А замінив всі від’ємні елементи матриці B.

Матриця B[7][7]:

| -88 -88 -88 -88 -88 -88 -88 |

| -88 48 -88 79 -88 39 64 |

| 31 34 -88 -88 0 -88 -88 |

| 82 3 -88 -88 10 -88 -88 |

| -88 23 10 -88 58 -88 54 |

| 68 55 22 -88 48 -88 -88 |

| 6 54 88 -88 -88 -88 -88 |

Кожний i-й елемент лінійного вектора С являє собою суму елементів i-го рядка матриці А.

С = | -28 -96 173 89 49 142 |

Мінімальне значення матриці А[6][6] = -88, де цей елемент в матриці А[6][6].

Матриця А[6][6] після упорядкування за зростанням значення елементів стовпця, де знаходиться мінімальний елемент матриці.

Матриця A[6][6]:

| 8 76 30 62 54 -88 |

| -80 -2 60 -81 90 -83 |

| 97 68 -13 47 53 -79 |

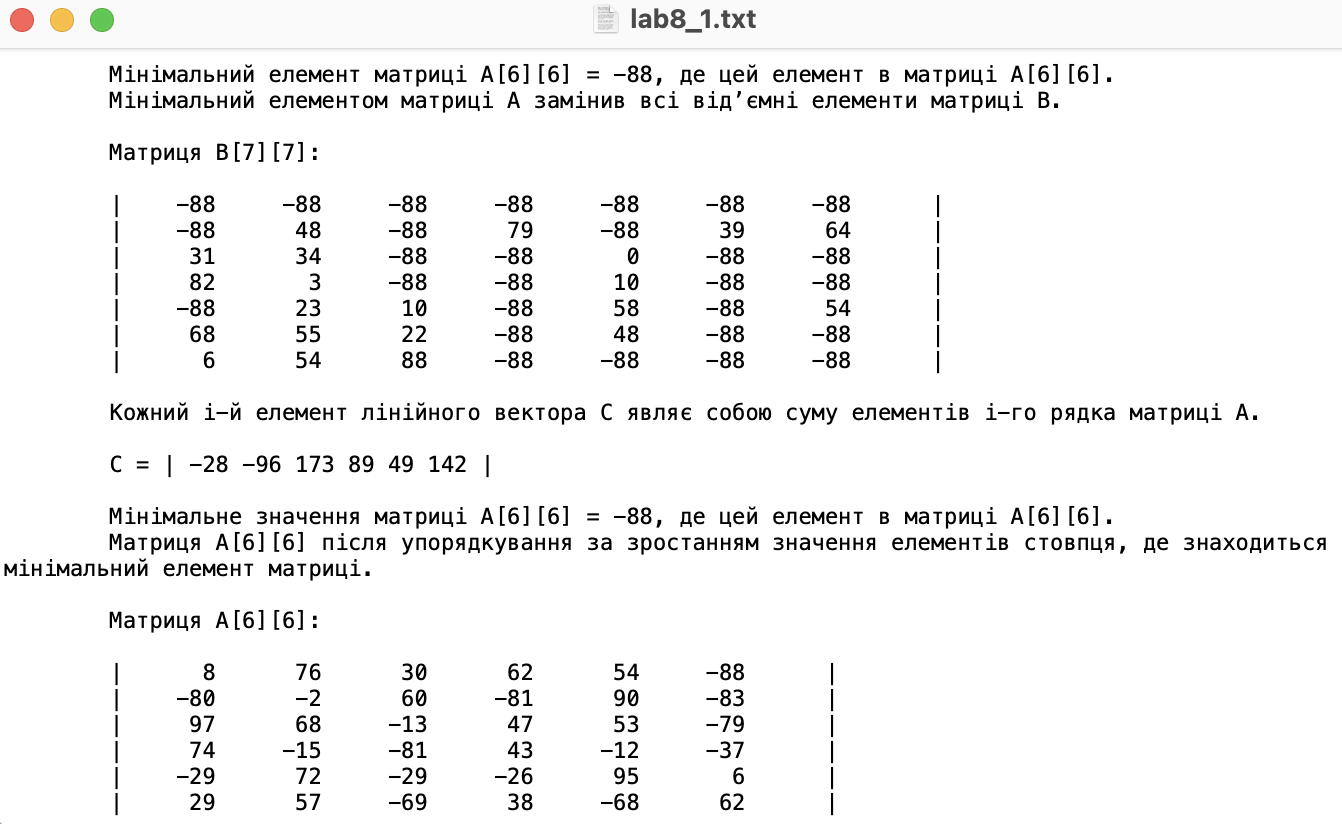
| 74 -15 -81 43 -12 -37 |

| -29 72 -29 -26 95 6 |

| 29 57 -69 38 -68 62 |

Записати дані у файл? (y/n) y

Введіть імʼя файлу (повне імʼя файлу): lab8\_1.txt



**Приклади заповнення з файлу**

Використаємо дані з “lab8\_2\_1.txt”:

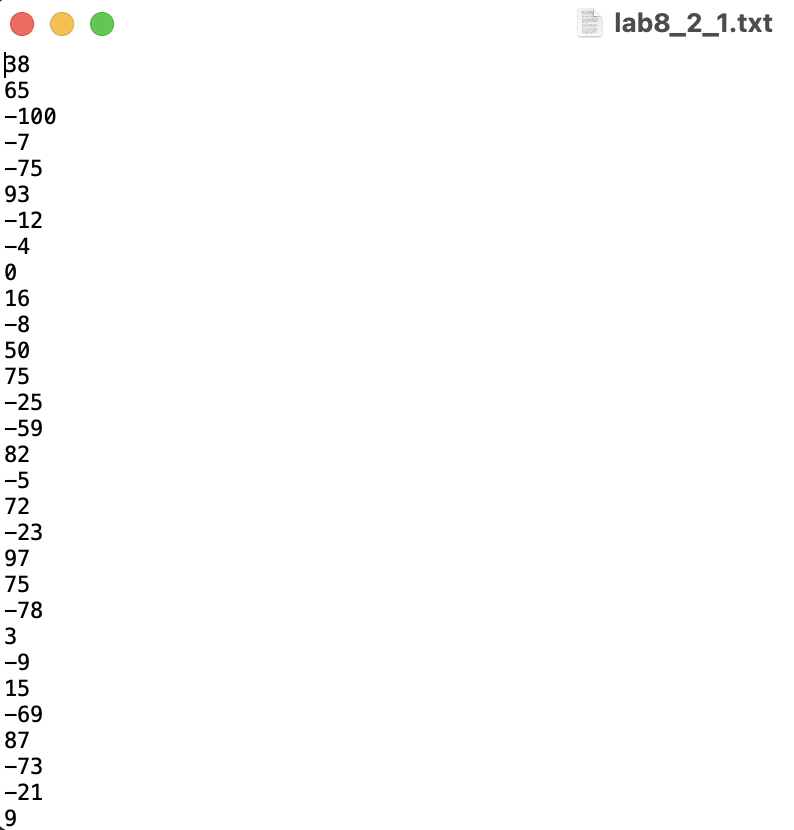


Рисунок 1 lab8\_2\_1.txt файл

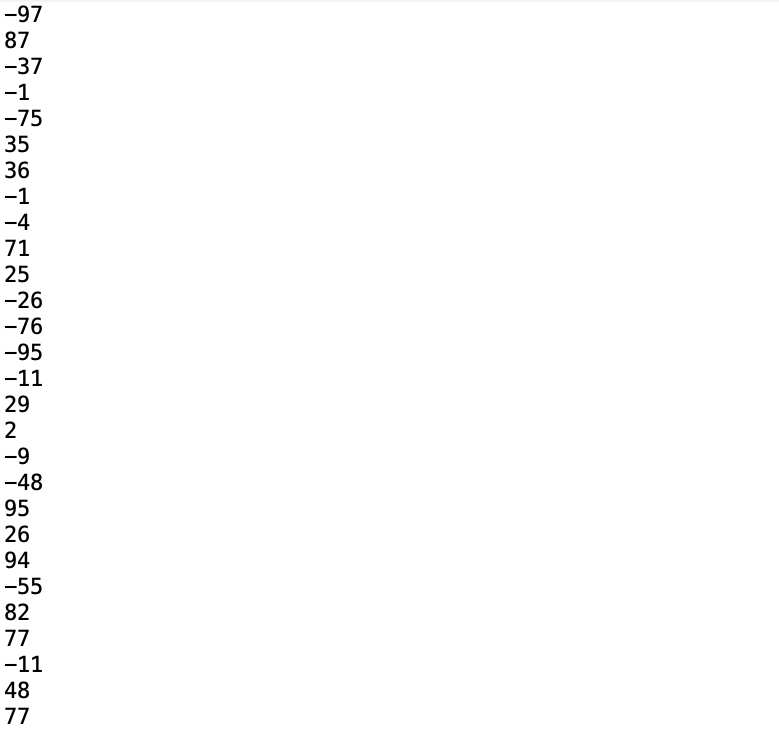


Рисунок 2 lab8\_2\_1.txt файл



Рисунок 3 lab8\_2\_1.txt файл

**Приклад 2.1 (без запису у файл)**

Mатриці А та В розміром n x m, де n і m ≥ 6.

Розмірність матриці А[n][m]:

n = 6

m = 6

Розмірність матриці B[n][m]:

n = 7

m = 7

Заповнити матрицю із файлу? (y/n) y

Введіть імʼя файлу (повне імʼя файлу): lab8\_2\_1.txt

Мінімальний елемент матриці А[6][7] = -100, де цей елемент в матриці А[1][3].

Mінімальний елементом матриці А замінив всі від’ємні елементи матриці B.

Матриця B[7][7]:

| 36 -100 -100 71 25 -100 -100 |

| -100 -100 29 2 -100 -100 95 |

| 26 94 -100 82 77 -100 48 |

| 77 -100 48 27 -100 74 -100 |

| 14 85 46 56 52 73 20 |

| 56 -100 27 99 87 42 -100 |

| 15 -100 61 -100 67 20 10 |

Кожний i-й елемент лінійного вектора С являє собою суму елементів i-го рядка матриці А.

С = | 14 42 140 65 -52 -88 |

Мінімальне значення матриці А[6][6] = -100, де цей елемент в матриці А[1][3].

Матриця А[6][6] після упорядкування за зростанням значення елементів стовпця, де знаходиться мінімальний елемент матриці.

Матриця A[6][6]:

| 38 65 -100 -7 -75 93 |

| 75 -25 -59 82 -5 72 |

| -97 87 -37 -1 -75 35 |

| -12 -4 0 16 -8 50 |

| -23 97 75 -78 3 -9 |

| 15 -69 87 -73 -21 9 |

Записати дані у файл? (y/n) n

**Приклад 2.2 (із записом у файл “lab8\_2\_2.txt”)**

Mатриці А та В розміром n x m, де n і m ≥ 6.

Розмірність матриці А[n][m]:

n = 6

m = 6

Розмірність матриці B[n][m]:

n = 7

m = 7

Заповнити матрицю із файлу? (y/n) y

Введіть імʼя файлу (повне імʼя файлу): lab8\_2\_1.txt

Мінімальний елемент матриці А[6][7] = -100, де цей елемент в матриці А[1][3].

Mінімальний елементом матриці А замінив всі від’ємні елементи матриці B.

Матриця B[7][7]:

| 36 -100 -100 71 25 -100 -100 |

| -100 -100 29 2 -100 -100 95 |

| 26 94 -100 82 77 -100 48 |

| 77 -100 48 27 -100 74 -100 |

| 14 85 46 56 52 73 20 |

| 56 -100 27 99 87 42 -100 |

| 15 -100 61 -100 67 20 10 |

Кожний i-й елемент лінійного вектора С являє собою суму елементів i-го рядка матриці А.

С = | 14 42 140 65 -52 -88 |

Мінімальне значення матриці А[6][6] = -100, де цей елемент в матриці А[1][3].

Матриця А[6][6] після упорядкування за зростанням значення елементів стовпця, де знаходиться мінімальний елемент матриці.

Матриця A[6][6]:

| 38 65 -100 -7 -75 93 |

| 75 -25 -59 82 -5 72 |

| -97 87 -37 -1 -75 35 |

| -12 -4 0 16 -8 50 |

| -23 97 75 -78 3 -9 |

| 15 -69 87 -73 -21 9 |

Записати дані у файл? (y/n) y

Введіть імʼя файлу (повне імʼя файлу): lab8\_2\_2.txt

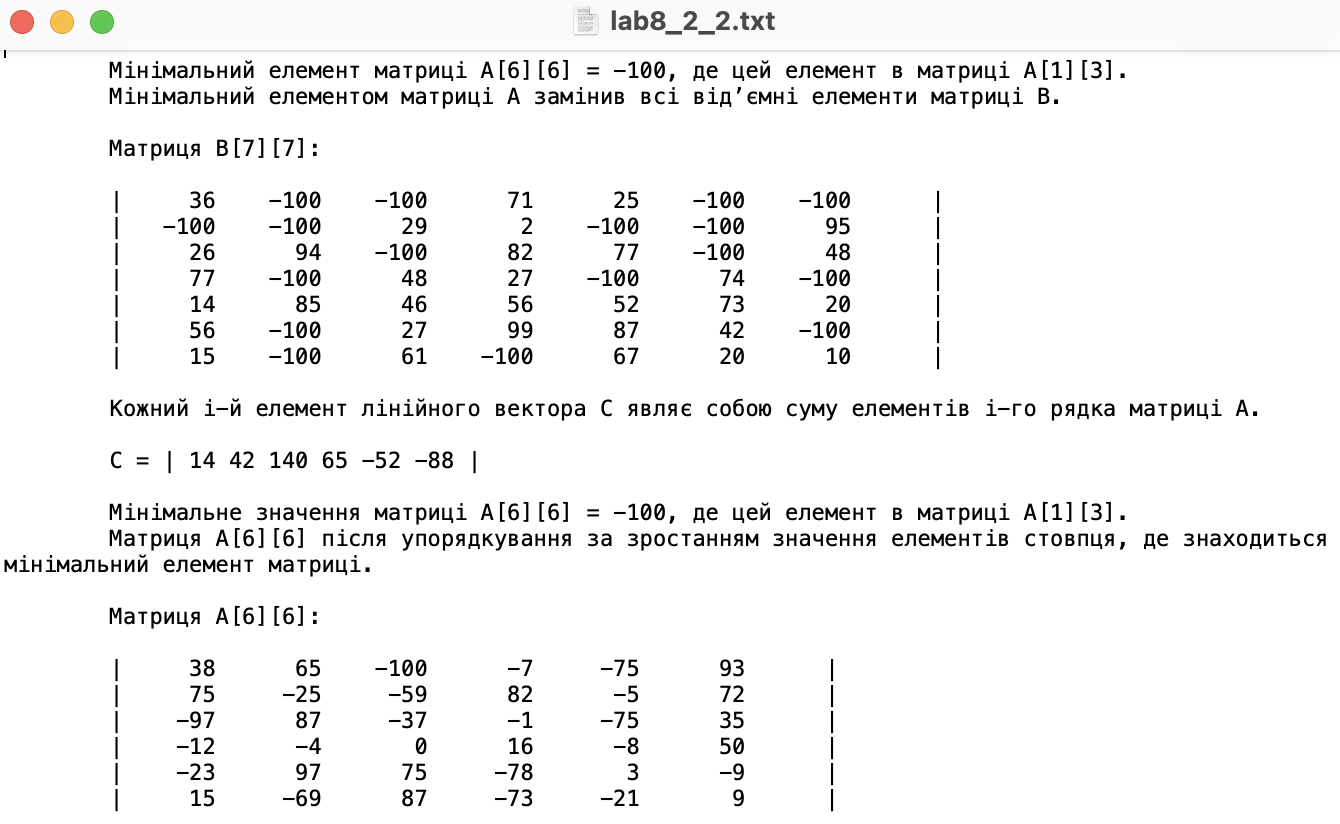


Рисунок 4 lab8\_2\_2.txt файл

**Додаток**

**Код програми**

#include <iostream>

#include <iomanip>

#include <fstream>

using namespace std;

void fillMatrixFromFile(int\*\* matrix1, int rows1, int cols1, int\*\* matrix2, int rows2, int cols2, const char\* filename)

{

ifstream file(filename);

if(file.peek() == ifstream::traits\_type::eof())

{

cerr << "\tФайл пустий.\n";

file.close();

return;

}

if (!file.is\_open())

{

cerr << "\tПомилка відкриття файлу.\n";

return;

}

if(!file.good())

{

cerr << "\tПомилка переведення данних із файлу.\n";

file.close();

return;

}

for (int i = 0; i < rows1; i++)

{

for (int j = 0; j < cols1; j++)

{

if (!(file >> matrix1[i][j]))

{

cerr << "\tНекорректний формат данних у файлі.\n";

file.close();

return;

}

}

}

for (int i = 0; i < rows2; i++)

{

for (int j = 0; j < cols2; j++)

{

if (!(file >> matrix2[i][j]))

{

cerr << "\tНекорректний формат данних у файлі.\n";

file.close();

return;

}

}

}

if (!file.eof())

{

cerr << "\tНевдалося прочитати всі дані з файлу.\n";

file.close();

return;

}

for (int i = 0; i < rows1; i++)

{

for (int j = 0; j < cols1; j++)

{

file >> matrix1[i][j];

}

}

for (int i = 0; i < rows2; i++)

{

for (int j = 0; j < cols2; j++)

{

file >> matrix2[i][j];

}

}

cout << "\tМатриця A[" << rows1 << "][" << cols1 << "]:\n\n";

for (int i=0; i<rows1; i++)

{

cout << "\t|";

for (int j=0; j<cols1; j++)

{

cout << setw(7) << matrix1[i][j] << " ";

}

cout << setw(7) << "|\n";

}

cout << "\n\n\tМатриця B[" << rows2 << "][" << cols2 << "]:\n\n";

for (int i=0; i<rows2; i++)

{

cout << "\t|";

for (int j=0; j<cols2; j++)

{

cout << setw(7) << matrix2[i][j] << " ";

}

cout << setw(7) << "|\n";

}

file.close();

}

void fillrandMatrix(int \*\*A, int n, int m)

{

for (int i=0; i<n; i++)

{

for (int j=0; j<m; j++)

{

A[i][j] = rand()%201-100;

}

}

}

void coutMatrix(int \*\*A, int n, int m)

{

for (int i=0; i<n; i++)

{

cout << "\t|";

for (int j=0; j<m; j++)

{

cout << setw(7) << A[i][j] << " ";

}

cout << setw(7) << "|\n";

}

cout << "\n\n";

}

void printToFile(int \*\*matrix1, int rows1, int cols1, int \*\*matrix2, int rows2, int cols2, const char\* filename, int o1, int o2, int min, int\* matrix3)

{

ofstream file(filename);

if (!file.is\_open())

{

cerr << "\tПомилка відкриття файлу.\n";

return;

}

file << "\n\tМінімальний елемент матриці А[" << rows1 << "][" << cols1 << "] = " << min << ", де цей елемент в матриці А[" << o1 + 1 << "][" << o2 + 1 << "].\n\t" << "Мінімальний елементом матриці А замінив всі від’ємні елементи матриці B.\n\n\tМатриця B[" << rows2 << "][" << cols2 << "]:\n\n";

for (int i = 0; i < rows2; i++)

{

file << "\t|";

for (int j = 0; j < cols2; j++)

{

file << setw(7) << matrix2[i][j] << " ";

}

file << setw(7) << "|\n";

}

file << "\n\tКожний i-й елемент лінійного вектора С являє собою суму елементів i-го рядка матриці А.\n\n\tС = | ";

for (int i = 0; i < rows1; i++)

{

file << matrix3[i] << " ";

}

file << "|\n\n\tМінімальне значення матриці А[" << rows1 << "][" << cols1 << "] = " << min << ", де цей елемент в матриці А[" << o1 + 1 << "][" << o2 + 1 << "]." << "\n\tМатриця А[" << rows1 << "][" << cols1 << "] після упорядкування за зростанням значення елементів стовпця, де знаходиться мінімальний елемент матриці.\n\n\tМатриця A[" << rows1 << "][" << cols1 << "]:\n\n";

for (int i = 0; i < rows1; i++)

{

file << "\t|";

for (int j = 0; j < cols1; j++)

{

file << setw(7) << matrix1[i][j] << " ";

}

file << setw(7) << "|\n";

}

file.close();

}

int main(int argc, char \*argv[])

{

srand(time(nullptr));

int n1, n2, m1, m2;

cout << "\n\tMатриці А та В розміром n x m, де n і m ≥ 6.\n\n\tРозмірність матриці А[n][m]:\n\n";

do

{

cout << "\tn = ";

cin >> n1;

if(n1<6)

{

cout << "\n\tОДЗ: n ≥ 6\n";

}

}while(n1<6);

cout << "\n";

do

{

cout << "\tm = ";

cin >> m1;

if(m1<6)

{

cout << "\n\tОДЗ: m ≥ 6\n";

}

}while(m1<6);

cout << "\n\tРозмірність матриці B[n][m]:\n\n";

do

{

cout << "\tn = ";

cin >> n2;

if(n2<6)

{

cout << "\n\tОДЗ: n ≥ 6\n";

}

}while(n2<6);

cout << "\n";

do

{

cout << "\tm = ";

cin >> m2;

if(m2<6)

{

cout << "\n\tОДЗ: m ≥ 6\n";

}

}while(m2<6);

int \*\*A = new int\*[n1]; for (int i=0; i<n1; i++) A[i] = new int[m1];

int \*\*B = new int\*[n2]; for (int i=0; i<n2; i++) B[i] = new int[m2];

int\* C = new int[n1]; delete[] C;

bool x = false;

char choice;

cout << "\n\tЗаповнити матрицю із файлу? (y/n) ";

while (x == false)

{

cin >> choice;

switch(choice)

{

case 'y':

x = true;

break;

case 'n':

x = true;

break;

default:

cout << "\tА я такого не казав вводити.\nСпробуйте ще раз.\n";

break;

}

}

cout << "\n\n";

if (choice == 'y')

{

x = false;

cout << "\n\tІмʼя файлу задано в командному рядку? (y/n) ";

while (x == false)

{

cin >> choice2;

switch(choice2)

{

case 'y':

x = true;

break;

case 'n':

x = true;

break;

default:

cout << "\tА я такого не казав вводити.\nСпробуйте ще раз.\n";

break;

}

}

if (choice2 == 'y')

{

if (argc == 2)

{

filename = argv[1];

}

else

{

cout << "\tІмʼя файлу не задано. Введіть імʼя файлу (повне імʼя файлу): ";

cin >> filename[100];

}

}

if (choice2 == 'n')

{

cout << "\tВведіть імʼя файлу (повне імʼя файлу): ";

cin >> filename[100];

}

fillMatrixFromFile(A, n1, m1, B, n2, m2, filename);

}

if (choice == 'n')

{

fillrandMatrix(A, n1, m1);

fillrandMatrix(B, n2, m2);

cout << "\tМатриця A[" << n1 << "][" << m1 << "]:\n\n";

coutMatrix(A, n1, m1);

cout << "\tМатриця B[" << n2 << "][" << m2 << "]:\n\n";

coutMatrix(B, n2, m2);

}

int min = A[0][0];

int o1 = 0, o2 = 0;

for (int i=0; i<n1; i++)

{

for (int j=0; j<m1; j++)

{

if (A[i][j]< min)

{

min = A[i][j];

o1 = i;

o2 = j;

}

}

}

for (int i=0; i<n2; i++)

{

for (int j=0; j<m2; j++)

{

if (B[i][j]< 0)

{

B[i][j] = min;

}

}

}

cout << "\tМінімальний елемент матриці А[" << n1 << "][" << m2 << "] = " << min << ", де цей елемент в матриці А[" << o1+1 << "][" << o2+1 << "].\n\t" << "Mінімальний елементом матриці А замінив всі від’ємні елементи матриці B.\n\n\tМатриця B[" << n2 << "][" << m2 << "]:\n\n";

coutMatrix(B, n2, m2);

cout << "\tКожний i-й елемент лінійного вектора С являє собою суму елементів i-го рядка матриці А.\n\n\tС = | ";

for (int i=0; i<n1; i++)

{

int sum = 0;

for (int j=0; j<m1; j++)

{

sum += A[i][j];

}

C[i] = sum;

cout << C[i] << " ";

}

cout << "|\n\n";

for (int i =0; i<n1-1; i++)

{

for(int j=i+1; j<n1; j++)

{

if (A[i][o2]>A[j][o2])

{

for (int k=0; k<m1; k++)

{

swap(A[i][k], A[j][k]);

}

}

}

}

cout << "\tМінімальне значення матриці А[" << n1 << "][" << m1 << "] = " << min << ", де цей елемент в матриці А[" << o1+1 << "][" << o2+1 << "]." << "\n\tМатриця А[" << n1 << "][" << m1 << "] після упорядкування за зростанням значення елементів стовпця, де знаходиться мінімальний елемент матриці.\n\n\tМатриця A[" << n1 << "][" << m1 << "]:\n\n";

coutMatrix(A, n1, m1);

x = false;

cout << "\tЗаписати дані у файл? (y/n) ";

while (x == false)

{

cin >> choice;

switch(choice)

{

case 'y':

x = true;

break;

case 'n':

x = true;

break;

default:

cout << "\tА я такого не казав вводити.\n\tСпробуйте ще раз.\n";

break;

}

}

if (choice == 'y')

{

char outFilename[100];

cout << "\tВведіть імʼя файлу (повне імʼя файлу): ";

cin >> outFilename;

printToFile(A, n1, m1, B, n2, m2, outFilename, o1, o2, min, C);

}

cout << "\n\n";

return 0;

}