Міністерство освіти і науки України

Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського"

Факультет інформатики та обчислювальної техніки Кафедра інформатики та програмної інженерії

Звіт

з лабораторної роботи № 8 з дисципліни

«Основи програмування 1. Базові конструкції»

«Багатовимірні масиви»

Варіант №11

Виконав студент ІП-14 Кирилюк Костянтин Віталійович

(шифр, прізвище, ім'я, по батькові)

Перевірив

( прізвище, ім'я, по батькові)

Київ 2021

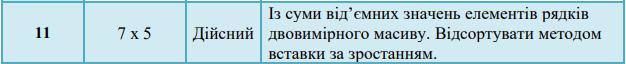
**Лабораторна робота №8**

**Тема**: Багатовимірні масиви.

**Мета**: опанувати технологію використання двовимірних масивів даних (матриць), навчитися розробляти алгоритми та програми із застосуванням матриць.

**Хід роботи**

**Задача.**



*Постановка задачі*

Результатом виконання задачі є відсортований за методом вставки масив сум негативних елементів рядків матриці.

**Випробування коду на С++:**

#include <iostream>

double\*\* input\_matrix(int n, int m);

void print\_matrix(double\*\* mat, int n, int m);

int find\_max\_col(double\*\* mat, int n, int m);

int find\_min\_col(double\*\* mat, int n, int m);

void switch\_col(double\*\* mat, int n, int m);

int main()

{

int n, m;

std::cout << "Enter n: ";

std::cin >> n;

std::cout << "Enter m: ";

std::cin >> m;

std::cout << "Enter C matrix: \n";

double\*\* mat1 = input\_matrix(n, m);

std::cout << "Enter B matrix: \n";

double\*\* mat2 = input\_matrix(n, m);

switch\_col(mat1, n, m);

switch\_col(mat2, n, m);

std::cout << "Y matrix: \n";

print\_matrix(mat1, n, m);

std::cout << "Z matrix: \n";

print\_matrix(mat2, n, m);

}

double\*\* input\_matrix(int n, int m) {

double\*\* mat = new double\* [n];

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

double\* arr = new double[m];

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

std::cin >> arr[j];

mat[i] = arr;

}

return mat;

}

void print\_matrix(double\*\* mat, int n, int m) {

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

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

std::cout << mat[i][j] << ' ';

}

std::cout << "\n";

}

}

int find\_max\_col(double\*\* mat, int n, int m) {

int max\_num = mat[0][0], col = 0;

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

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

if (max\_num < mat[i][j]) {

max\_num = mat[i][j];

col = j;

}

}

}

return col;

}

int find\_min\_col(double\*\* mat, int n, int m) {

int min\_num = mat[0][0], col = 0;

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

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

if (min\_num > mat[i][j]) {

min\_num = mat[i][j];

col = j;

}

}

}

return col;

}

void switch\_col(double\*\* mat, int n, int m) {

int first\_col = find\_max\_col(mat, n, m);

int second\_col = find\_min\_col(mat, n, m);

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

int temp = mat[i][first\_col];

mat[i][first\_col] = mat[i][second\_col];

mat[i][second\_col] = temp;

}

}

**Результат:**

**Висновок:**

Під час даної лабораторної роботи були досліджені підпрограми як програмні фрагменти. Було використано динамічний двовимірний масив, головною особливістю якого є можливість вказання к-сті елементів користувачем. Були набуті навички їх використання у програмних специфікаціях. Була побудована математична модель. Для обчислення виразу була створена елементарна програма. Результати програми виявилися правильними, що стверджує на її дієвість. Завдання було виконано на мові програмування c++.