# 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 20

Виконав студент Лошак Віктор Іванович

(шифр, прізвище, ім'я, по батькові)

Перевірив

( прізвище, ім'я, по батькові)

Київ 2021

**Лабораторна робота 8**

**Мета:**

Опанувати технологію використання двовимірних масивів даних(матриць), навчитися розробляти алгоритми та програми із застосування матриць.

**Тема:**

Задані цілочисельні матриці A(m\*n) та M(m\*n), що містять від’ємні елементи. Нехай B(n) і T(n)- середні арифметичні елементів стовпців заданих матриць. Обчислити

**Математична модель:**

Для обчислення виразу при заданих масивах A(m\*n) та M(m\*n), використаємо підпрограми. Для спрощення обрахунків використаємо чотири підпрограми. Оскільки за умовою користувач сам повинен вказати розмірність матриці то під час компіляції розмір матриць не може бути визначеним, отже для ініціалізації заданих структур використаємо динамічне виділення пам’яті. Перша підпрограма буде проводити виділення пам’яті для двовимірного масиву заданого розміру і повертати вказівник на нього. Друга підпрограма буде розраховувати середнє арифметичне елементів переданого їй стовпчика матриці і повертати його. Третя функція використовується для спрощення логіки виконання основного циклу програми а конкретніше обчислення суми всіх доданків заданої системи. Ця функція повертає ціле значення результату. Четверта функція викликається після завершення основного циклу і відповідає за правильне видалення всіх динамічно виділених ресурсів, у нашому випадку двовимірних масивів. Аргументи передаватимемо як константи , якщо функція не повинна змінювати їх значень, в деяких випадкав можлива передача вказівників, щоб функція мала доступ до динамічно виділених масивів. Для ініціалізації масивів використаємо арифметичні цикли.

**1)**Приклад коду на С++:

#include <iostream>

using namespace std;

int\*\* matrixInit(const int rowLength,const int columnLength)

{

int\*\* matrix = new int\* [rowLength];

cout << "Enter the matrix using spaces to divide elements: " << endl;

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

{

matrix[i] = new int[columnLength];

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

{

cin >> matrix[i][j];

cin.ignore(1);

}

}

return matrix;

}

double columnArithmeticAverage(const int\*\* matrix, const int columnNum, const int rowLength)

{

double sum = 0;

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

{

sum += static\_cast<double>(matrix[i][columnNum]);

}

return (sum / static\_cast<double>(rowLength));

}

double mainCycle(const int\*\* matrix1,const int\*\* matrix2,const int rowLength,const int columnLength)

{

double sum = 0;

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

{

sum += abs(columnArithmeticAverage(matrix1, i, rowLength) - columnArithmeticAverage(matrix2, i, rowLength));

}

return sum;

}

void deleteMatrix(int \*\*matrix,int rowLength, int columnLength)

{

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

{

delete[] matrix[i];

}

delete[] matrix;

}

int main( )

{

// user enters the wanted 1size of matrix

int rowLength;

int columnLength;

std::cout << "Enter the type of matrixes you want to create(rows columns): ";

cin >> rowLength >> columnLength;

// creation of two dynamically allocated matrixes of the given size

int\*\* matrix1 = matrixInit(rowLength, columnLength);

int\*\* matrix2 = matrixInit(rowLength, columnLength);

//calculating the result according top the given formula and displaying it on the screen

cout << "The resulting sum is " << mainCycle(matrix1, matrix2, rowLength, columnLength)<<endl;

//deleting dynamically allocated two dimensional arrays

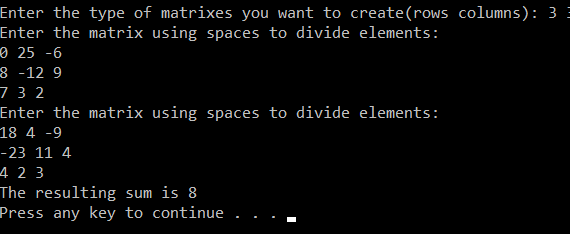
deleteMatrix(matrix1, rowLength, columnLength);

deleteMatrix(matrix2, rowLength, columnLength);

return 1;

}

Виконання коду на С++:



**Висновок:**

Отже за допомогою даного алгоритму ми успішно обрахували і відобразили на екрані значення заданої суми розрахованої за формулою при довільних ввідних даних від користувача, за допомогою використання підпрограм на компільованій мові С++ .Ми також створили механізми уникнення втрат пам’яті та аналізували умову для знаходження оптимальних шляхів виконання завдання.