# 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 05

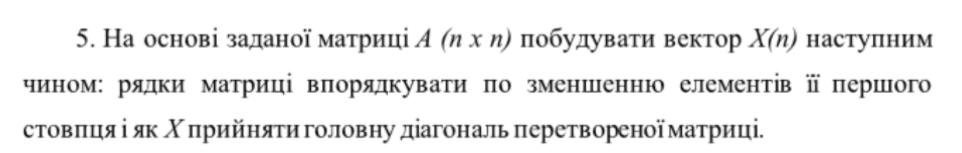
Виконав студент ІП-11 Гіжицький Даниїл Олександрович

(шифр, прізвище, ім'я, по батькові)

Перевірив

( прізвище, ім'я, по батькові)

Київ 2021

1. Мета:
2. 
3. Постановка задачі:

Ініціалізуємо масив, який надалі буде виконувати роль матриці. Заповнюємо масиви по-рядково. Після цього напишемо функцію, яка в якості аргументу буде приймати матрицю, , а повертати результат роботи буде впорядковану матрицю. Також напишемо функцію яка буде перетворювати головну діагональ на масив та виводити його. Математична модель:

|  |  |  |  |
| --- | --- | --- | --- |
| Номер | Назва | Тип | Призначення |
| 1 | matr | Int(n x n) | Користувацький ввід |
| 2 | n | int | Користувацький ввід |
| 3 | main\_diag | Int(n) | Результат |

Написана програма на мові C+

#include <iostream>

#include <ctime>

using namespace std;

int\*\* input\_matrix(int n);

int\*\* rand\_matrix(int n);

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

void print\_array(int n, int\* arr);

void sort(int n, int\*\* arr);

int\* main\_diagonal(int n, int\*\* matr);

int main()

{

srand(time(NULL));

int n;

cin >> n;

int\*\* matr = rand\_matrix(n);

print\_matrix(n, matr);

sort(n, matr);

print\_matrix(n, matr);

int\* main\_diag = main\_diagonal(n, matr);

print\_array(n, main\_diag);

}

int\*\* input\_matrix(int n) {

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

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

int\* arr = new int[n];

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

cin >> arr[j];

mat[i] = arr;

}

cout << '\n';

return mat;

}

int\*\* rand\_matrix(int n) {

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

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

int\* arr = new int[n];

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

arr[j] = rand() % 21 - 10;

mat[i] = arr;

}

cout << '\n';

return mat;

}

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

cout << "Matrix:\n";

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

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

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

}

cout << "\n";

}

cout << "\n";

}

void print\_array(int n, int\* arr) {

cout << "Array:\n";

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

cout << arr[i] << ' ';

}

cout << '\n';

}

void sort(int n, int\*\* arr) {

for (int i = 1; i < n; i++) {

int\* cur = arr[i];

int j = i - 1;

while (j >= 0 and cur[0] > arr[j][0]) {

arr[j + 1] = arr[j];

j -= 1;

arr[j + 1] = cur;

}

}

}

int\* main\_diagonal(int n, int\*\* matr) {

int\* arr = new int[n];

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

{

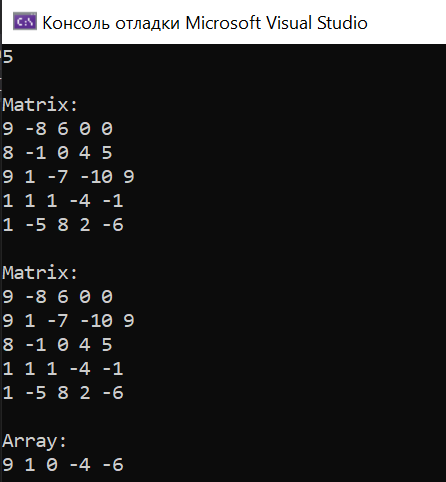
arr[i] = matr[i][i];

}

return arr;

}

Результати роботи програми



1. Висновок: В ході дослідження поставленої задачі, я придбав та відпрацював навички у роботі із звичайними та багатовимірними масивами і коректної роботи з ними.