

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

«Алгоритми та структури даних-1.

Основи алгоритмізації»

«Дослідження алгоритмів пошуку та сортування»

Варіант 1

Виконав студент ІП-15, Багачок Вадим Дмитрович

(шифр, прізвище, ім'я, по батькові)

Перевірив - Вечерковська А.С

(пріз
вище, ім'я, по батькові)

Київ 2021

Лабораторна робота 8
Дослідження алгоритмів пошуку та сортування

Мета – дослідити алгоритми пошуку та сортування, набути практичних навичок використання цих алгоритмів під час складання програмних специфікацій.

Варіант 1

№ варіанта	Розмірність	Тип даних	Обчислення значень елементів одновимірного масиву
1	5 x 5	Цілий	Із значень елементів головної діагоналі двовимірного масиву. Відсортувати обміном за зростанням.

Постановка задачі:

Потрібно розробити алгоритм та написати програму, яка складається з наступних дій:
На першому кроці потрібно описати змінну індексованого типу (**дтовимірний масив**) згідно з варіантом. На другому кроці потрібно ініціювати змінну, що описана в п.1 даного завдання. На третьому кроці потрібно створити нову змінну індексованого типу (**одновимірний масив**) та її ініціювати значеннями, що обчислюються згідно з варіантом (табл. 1).

Математична модель :

Спочатку створюємо функцію в якій і відбуватимуться всі обчислення. Всередині функції ініціалізуємо двовимірний **масив цілих чисел** розміром **5x5**. За допомогою двох циклів **for** заповнити масив рандомними значеннями. циклу **for**. Після того як ми заповнили двовимірний масив рандомними значеннями, створюємо одновимірний масив і заповнюємо його значеннями головної діагоналі двовимірного масиву. Потім за допомогою бульбашкового сортування сортуємо значення від меншого до великого. Після того як одновимірний масив буде відсортований ми виводимо його значення користувачеві.

Змінна	Тип	Ім'я	Призначення
array	ціле	array	Проміжні дані
newArray	ціле	newArray	Проміжні дані
i	ціле	i	Проміжні дані
j	ціле	j	Проміжні дані
tmp	ціле	tmp	Проміжні дані
size	ціле	size	Проміжні дані

chened	bool	chened	Проміжні дані
--------	------	--------	---------------

Псевдокод :

Основна програма :

```

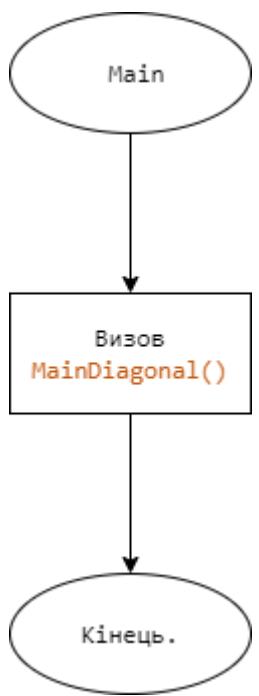
    Початок
    Виклик MainDiagonal()
    Кінець.
```

Підпрограма :

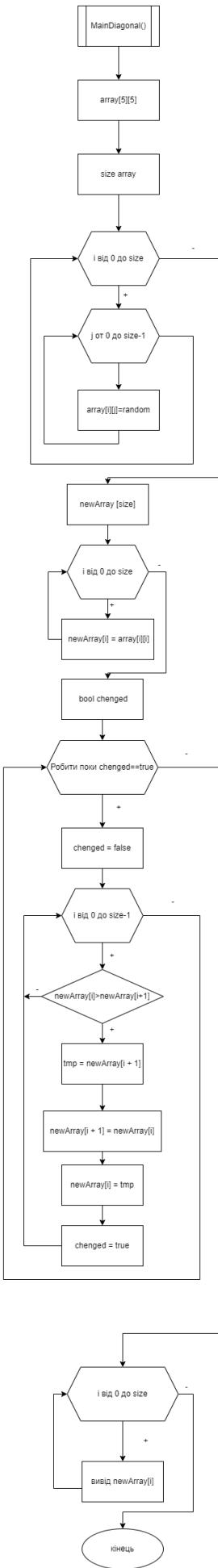
```

MainDiagonal()
    array[5][5]
    size=length(array)
    повторити для i від 0 до size
        повторити для j від 0 до size
            array[i][j]= rand
        все повторити
    все повторити
    newArray [size]

    повторити для i від 0 до size
        newArray[i] = array[i][i]
    все повторити
    chened= bool
    повторити :
        chened= false
        повторити для i від 0 до size-1
            якщо newArray[i]>newArray[i+1]
                то tmp = newArray[i + 1]
                newArray[i + 1] = newArray[i]
                newArray[i] = tmp
                chened = true
            все якщо
        все повторити
    поки chened== true
    повторити для i від 0 до size
        вивід newArray[i]
    все повторити
    кінець.
```



На наступній сторінці підпрограма →



```
#include <iostream>
#include<string>
using namespace std;
void MainDiagonal()
{
    int array[5][5];
    int size = sizeof(array) / sizeof(array[0]);
    for (int i = 0;i < size;i++)
    {
        for (int j = 0; j < size;j++)
        {
            array[i][j] = 0 + rand();
        }
    }
    int* newArray = new int[size];

    for (int i = 0;i < size;i++)
    {
        newArray[i] = array[i][i];
    }
    bool chenged;
    do
    {
        chenged = false;
        for (int i = 0;i < size -1;i++)
        {
            if (newArray[i]>newArray[i+1])
            {
                int tmp = newArray[i + 1];
                newArray[i + 1] = newArray[i];
                newArray[i] = tmp;
                chenged = true;
            }
        }
    } while (chenged==true);

    for (int i = 0;i < size;i++)
    {
        cout<< to_string(newArray[i])+"\\n";
    }
}
int main()
{
    MainDiagonal();
}

C++ программа
```

Скриншот Microsoft Visual Studio, показывающий окно редактора кода с функцией `MainDiagonal()` и окно консоли отладки.

```
// Lab-8 ASD.cpp : Этот файл содержит функцию "main". Здесь начинается и заканчивается выполнение программы.
//
#include <iostream>
#include<cstdlib>
using namespace std;
void MainDiagonal()
{
    int array[5][5];
    int size = sizeof(array) / sizeof(array[0]);
    for (int i = 0; i < size; i++)
    {
        for (int j = 0; j < size; j++)
        {
            array[i][j] = 0 + rand();
        }
    }
    int* newArray = new int[size];
    for (int i = 0; i < size; i++)
    {
        newArray[i] = array[i][i];
    }
    bool changed;
    do
    {
        changed = false;
        for (int i = 0; i < size - 1; i++)
        {
            if (newArray[i] > newArray[i + 1])
                newArray[i] = newArray[i + 1];
        }
    } while (changed);
    delete[] newArray;
}
```

Консоль отладки Microsoft Visual Studio

C:\Программы КПИ\С++\Lab-8 ASD\Debug\Lab-8 ASD.exe (процесс 21988) завершил работу с кодом 0.
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрывать консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно.

Вывод

Показать выходные данные из: Отладка

Поток 0x101e завершился с кодом 0 (0x0).
"Lab-8 ASD.exe" (Win32). Загружено "C:\Windows\SystemW64\kernel32.dll".
"Lab-8 ASD.exe" (Win32). Загружено "C:\Windows\SystemW64\msvcrt.dll".
Поток 0x2434 завершился с кодом 0 (0x0).
Программа "[21988] Lab-8 ASD.exe" завершилась с кодом 0 (0x0).

Список ошибок Вывод

Висновки: Ми дослідили алгоритми пошуку та сортування, набули практичних навичок використання цих алгоритмів під час складання програмних специфікацій.