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 25

Виконав ІП-15, Плугатирьов Дмитро Валерійович

студент (шифр, прізвище, ім’я, по батькові)

Перевірив Вєчерковська Анастасія Сергіївна

(прізвище, 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 25**

**Завдання**

Розробити алгоритм та написати програму, яка складається з наступних

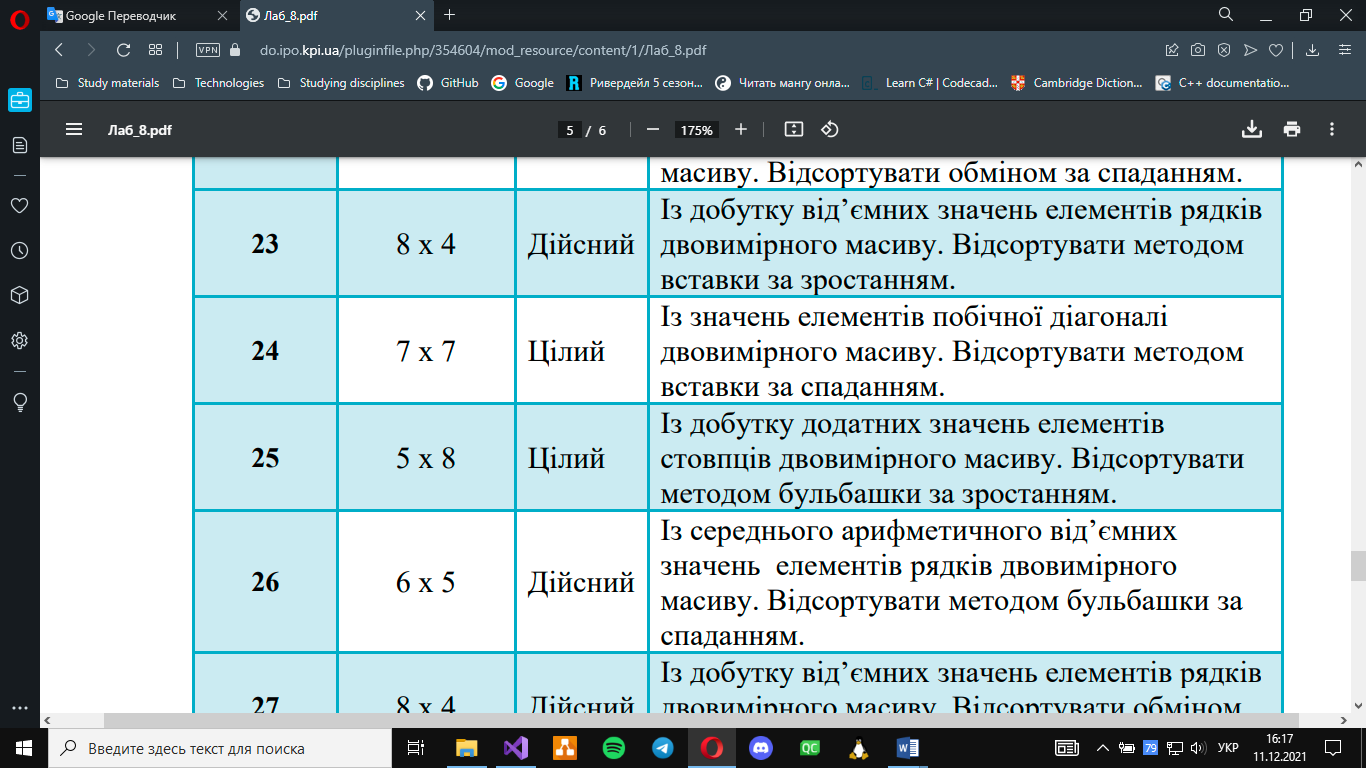
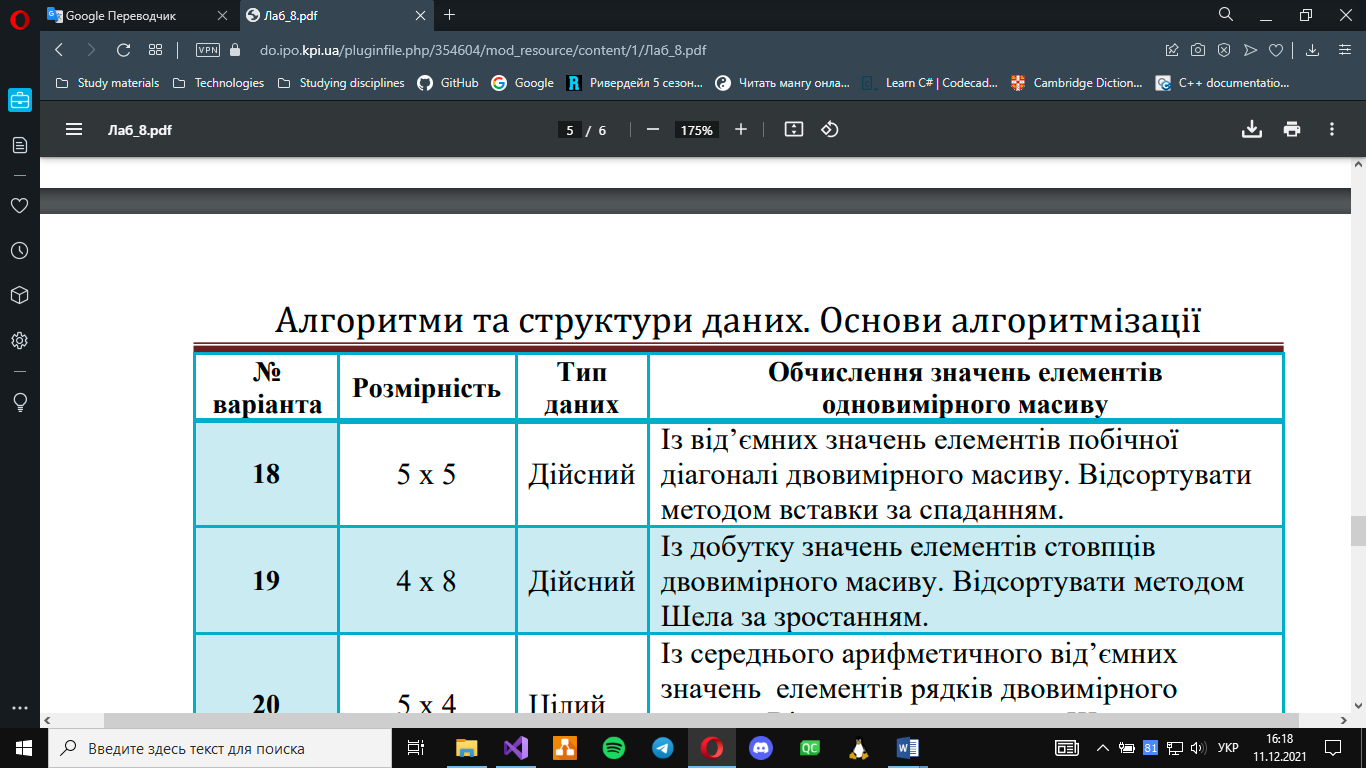
дій:

1. Опису змінної індексованого типу (двовимірний масив) згідно з варіантом.

2. Ініціювання змінної, що описана в п.1 даного завдання.

3. Створення нової змінної індексованого типу (одновимірний масив) та її

ініціювання значеннями, що обчислюються згідно з варіантом



**1. П о с т а н о в к а з а д а ч і**

Описати змінну індексованого типу, котра вміщує 5 елементів: послідовності змінних такого ж типу по вісім елементів кожна. Заповнити утворені послідовності випадковими цілими значеннями. Виконати добуток додатних елементів кожного стовпця утвореного двовимірного масиву, розміщуючи результати в новоствореній послідовності. Відсортувати цю послідовність методом бульбашки за зростанням.

**2. П о б у д о в а м а т е м а т и ч н о ї м о д е л і**

|  |  |  |  |
| --- | --- | --- | --- |
| *Змінна* | *Тип* | *Ім’я* | *Призначення* |
| Кількість рядків матриці | Цілочисельний | ROW | Початкові дані |
| Кількість колонок матриці | Цілочисельний | COL | Початкові дані |
| Матриця | Цілочисельний | arr[ROW][COL] | Початкові дані |
| Новоутворена послідовність | Цілочисельний | newArr[COL] | Початкові дані |
| Добуток додатних елементів стовпців | Підпрограма | mult\_cols | Проміжні дані |
| Заповнення матриці | Підпрограма | fill\_arr | Початкові дані |
| Сортування новоутвореної послідовності | Підпрограма | sort | Результат |
| Поточний індекс у новій послідовності | Цілочисельний | pos | Проміжні дані |
| Значення елемента послідовності | Цілочисельний | temp | Проміжні дані |
| Лічильник циклу | Цілочисельний | u | Проміжні дані |
| Лічильник циклу | Цілочисельний | i | Проміжні дані |

Робота з матрицею відбувається переважно з використанням арифметичних вкладених циклів. В тому числі, й обрахунок значень елементів нової послідовності з участю стовпців матриці шляхом перемножування їх додатних елементів. Сортування відбувається з ліва на право методом бульбашки за зростанням: створюється тимчасова змінна для забезпечення обміном значень між сусідніми елементами за умови більшості лівішого.

Дія х \*= і означає x := x \* і.

Дія x++ означає x := x + 1.

Дія x := rand() % num1 – num2 означає присвоєння згенерованого псевдовипадковим чином цілого числа до змінної x в межах від num1 – num2 - 1 до –num2.

**3. Р о з в ' я з а н н я**

Програмні специфікації записати у псевдокоді та графічній формі у вигляді блок-схеми.

*Крок 1*. Визначити основні дії.

*Крок 2*. Ініціалізація новоствореної послідовності добутками додатних елементів.

*Крок 3.* Сортування послідовності за зростанням.

**4. П с е в д о к о д**

**О с н о в н а п р о г р а м а:**

*Крок 1*

**початок**

ініціалізація новоствореної послідовності добутками додатних елементів

сортування послідовності за зростанням

**кінець**

*Крок 2*

**початок**

ROW := 5

COL := 8

**fill\_arr(arr, ROW, COL)**

**mult\_cols(arr, newArr, ROW, COL)**

сортування послідовності за зростанням

**кінець**

*Крок 3*

**початок**

ROW := 5

COL := 8

**fill\_arr(arr, ROW, COL)**

**mult\_cols(arr, newArr, ROW, COL)**

**sort(newArr, COL)**

**кінець**

**П і д п р о г р а м и:**

**fill\_arr(arr, ROW, COL)**

**повторити для** u **від** 0 **до** ROW

**повторити для** i **від** 0 **до** COL

arr[u][i] := rand() % 500 - 250

**все повторити**

**все повторити**

**кінець**

**mult\_cols(dArr, newArr, ROW, COL)**

pos := 0

**повторити для** i **від** 0 **до** COL

**повторити для** u **від** 0 **до** ROW

**якщо** dArr[u][i] > 0

**то**

newArr[pos] \*= dArr[u][i]

**все якщо**

**все повторити**

**якщо** newArr[pos] == 1

**то**

newArr[pos] := 0

**все якщо**

pos++

**все повторити**

**кінець**

**sort(arr, SIZE)**

**повторити для** u **від** 0 **до** SIZE

**повторити для** i **від** 0 **до** SIZE - 1

**якщо** arr[i] > arr[i + 1]

**то**

temp := arr[i]

arr[i] := arr[i + 1]

arr[i + 1] := temp

**все якщо**

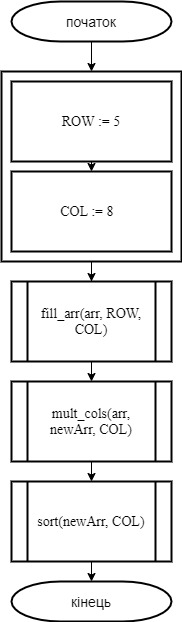
**все повторити**

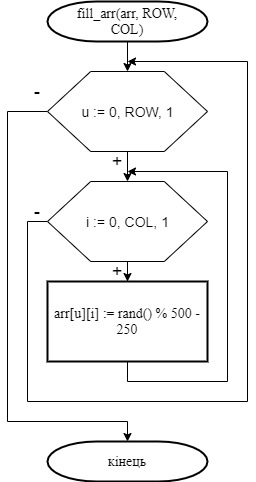
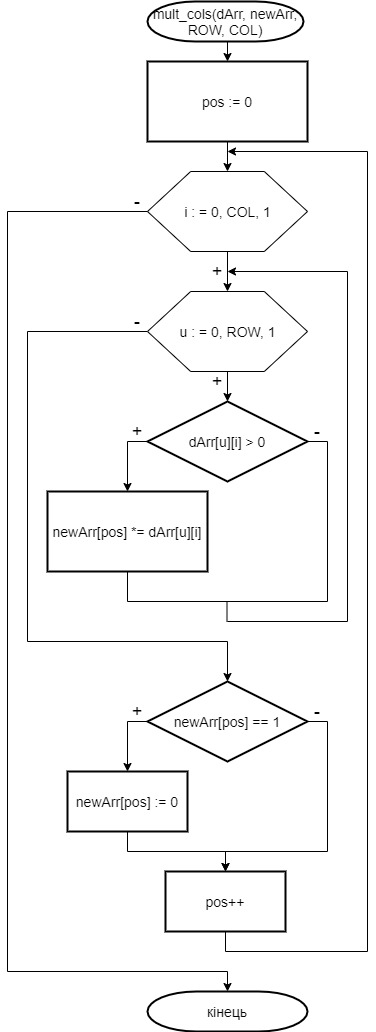
**все повторити**

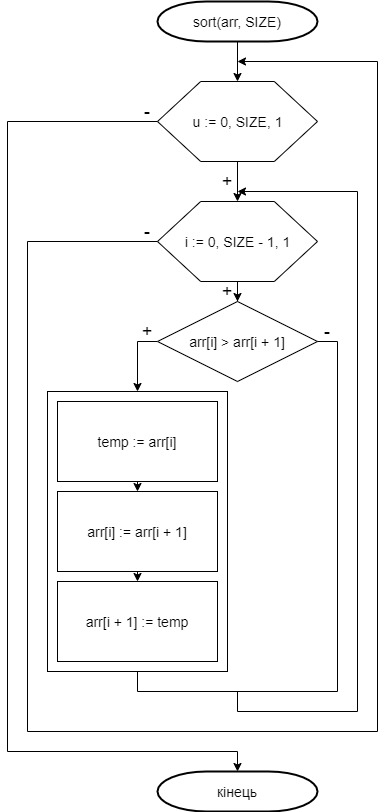
**кінець**

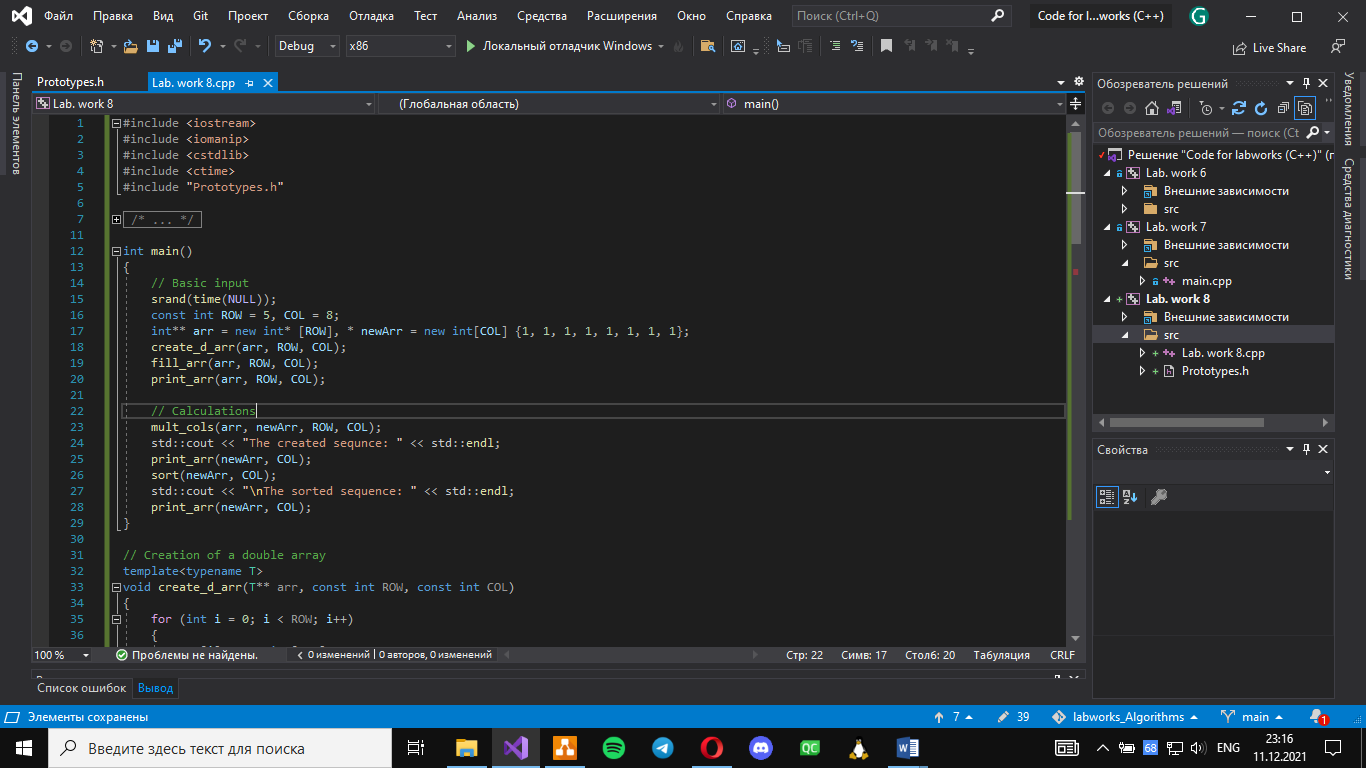
*Блок-схема*

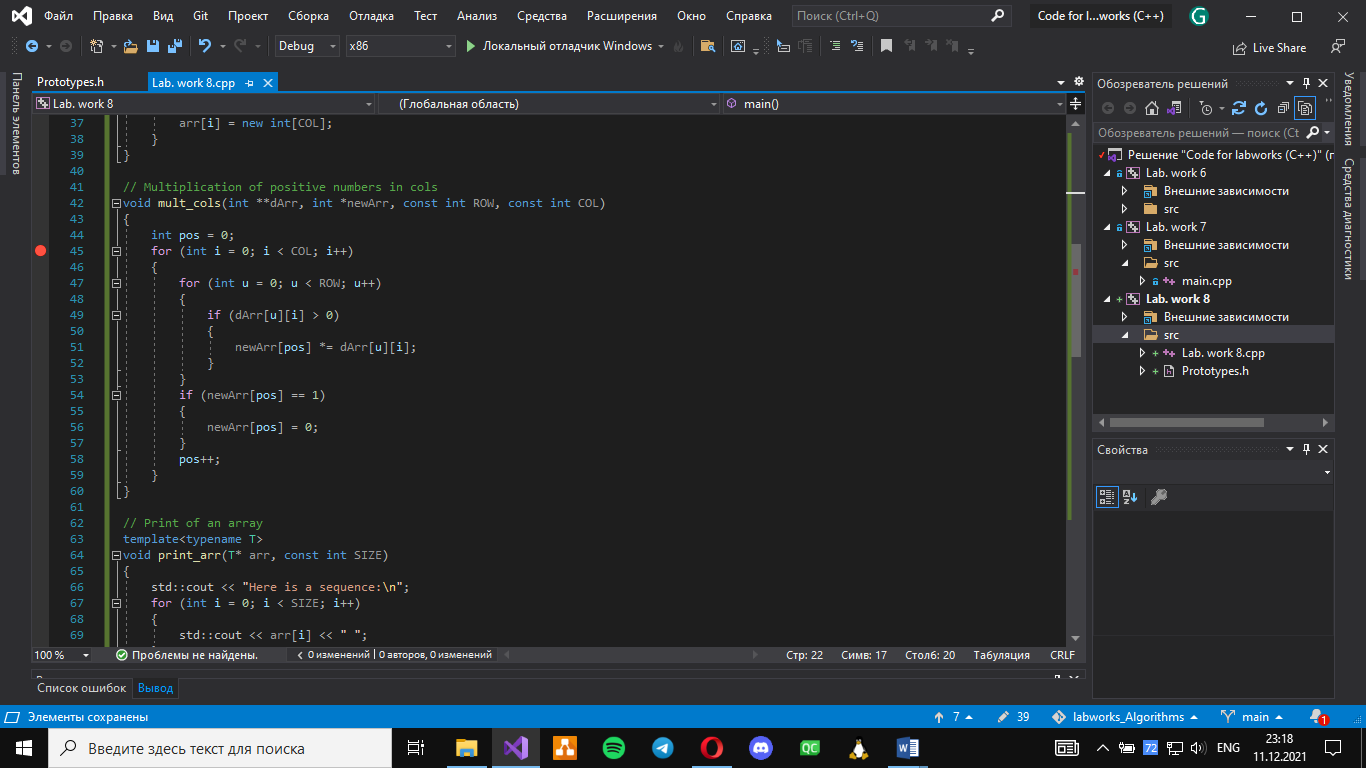
**О с н о в н а п р о г р а м а:**

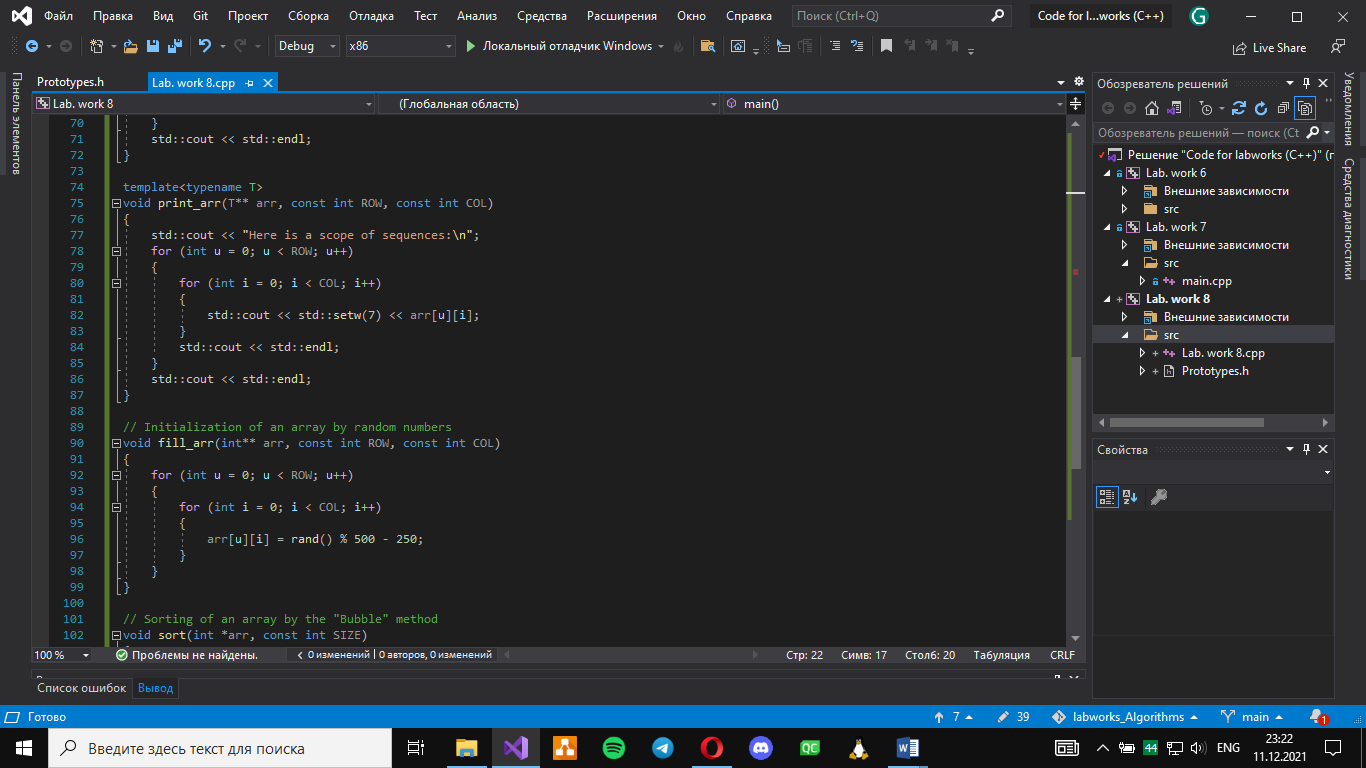
****

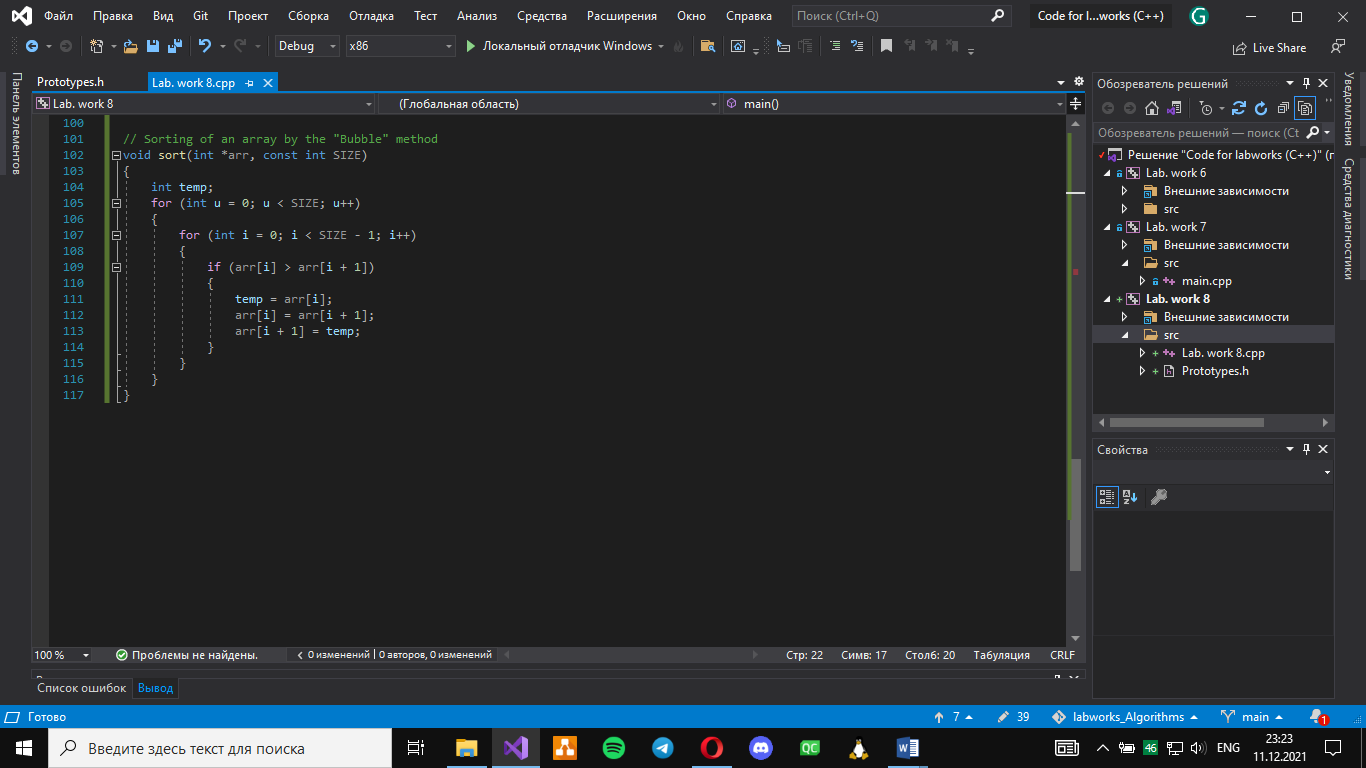
**П і д п р о г р а м и:**

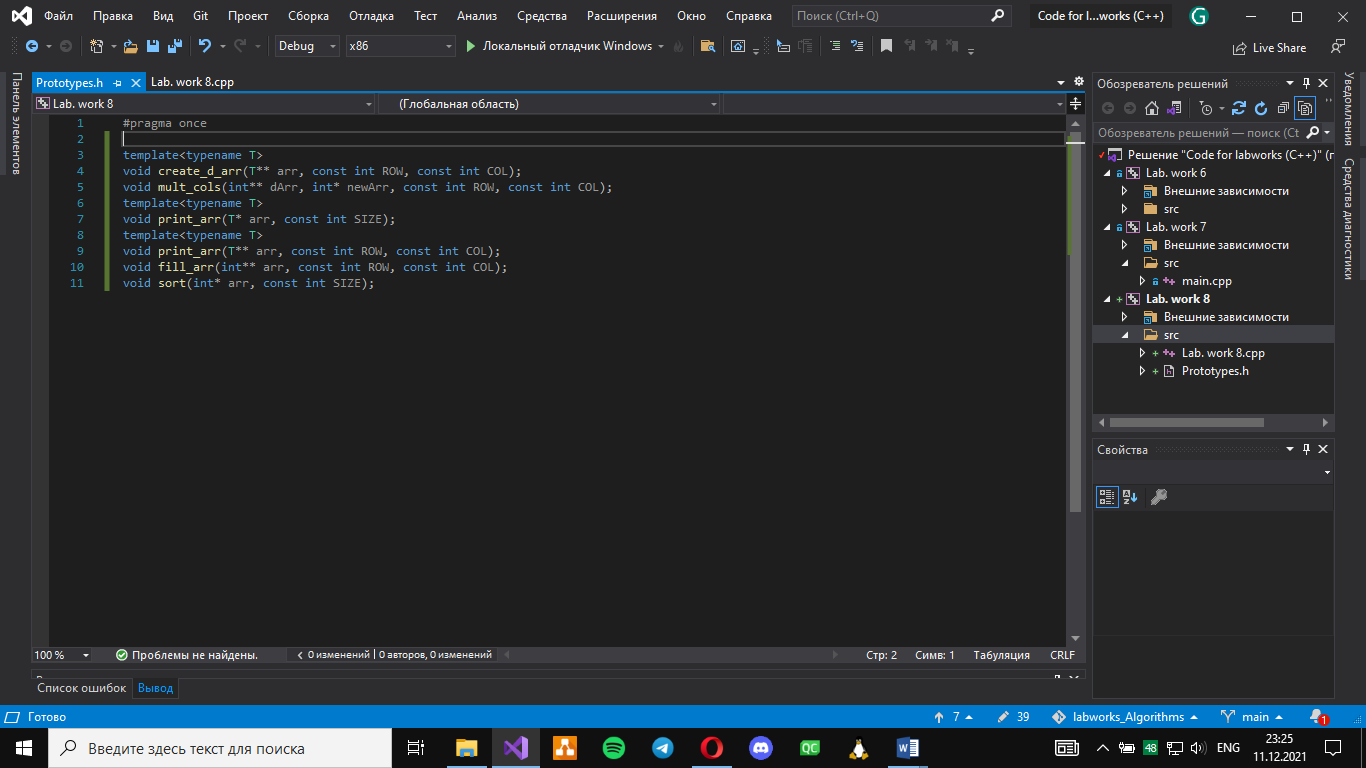


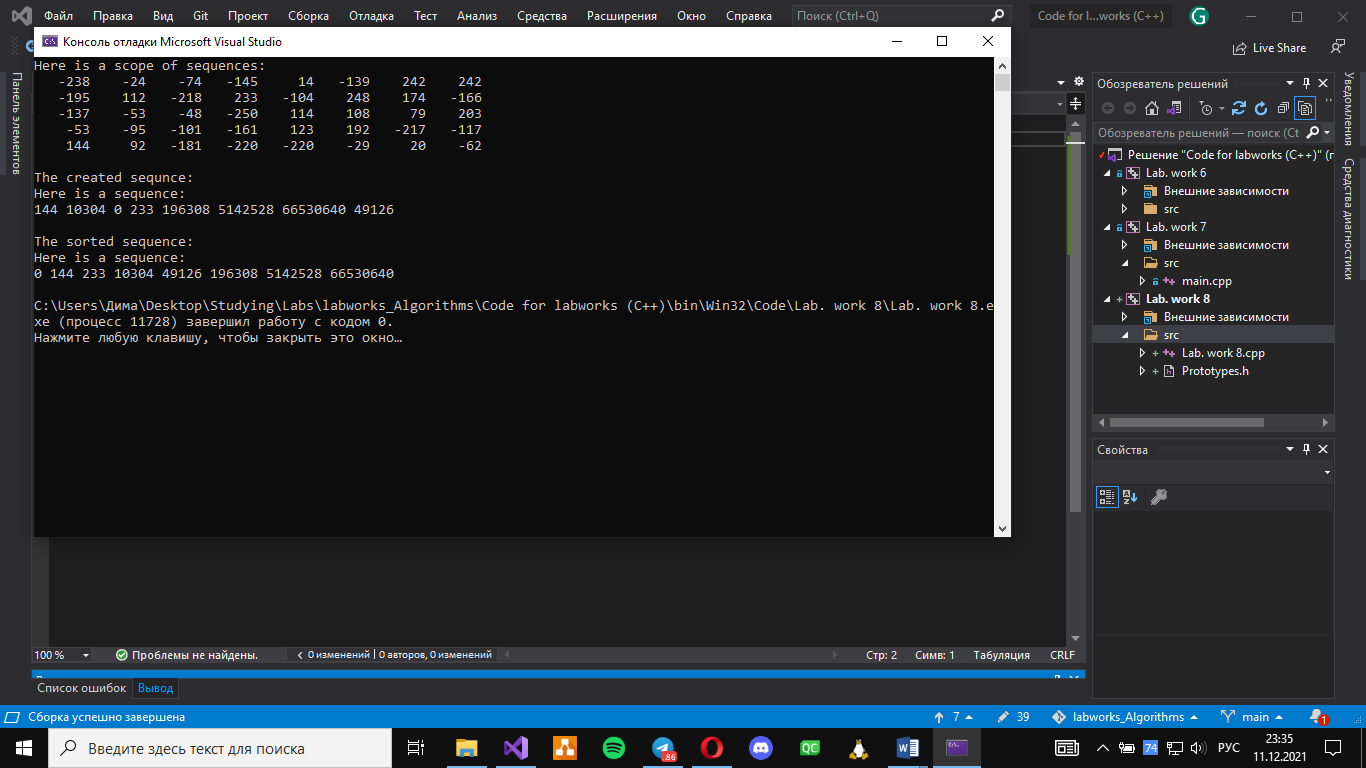
**5. К о д п р о г р а м и**









**Т е с т у в а н н я**

**6. В и с н о в о к**

На цій лабораторній роботі я дослідив алгоритми пошуку та сортування та набув практичних навичок використання цих алгоритмів під час складання програмних специфікацій. Мною був використаний метод «бульбашки» для сортування послідовності за зростанням.