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 № 7 з дисципліни

«Алгоритми та структури даних-1.

Основи алгоритмізації»

«Дослідження рекурсивних алгоритмів»

Варіант 20

Виконав студент Медвідь Олександр Русланович

(шифр, прізвище, ім'я, по батькові)

Перевірив Доцент кафедри ІПІ Мартинова О. П.

( прізвище, ім'я, по батькові)

Київ 202 1

**Лабораторна робота №7**

**Мета**: дослідити методи послідовного пошуку у впорядкованих і невпорядкованих послідовностях та набути практичних навичок їх використання під час складання програмних специфікацій.

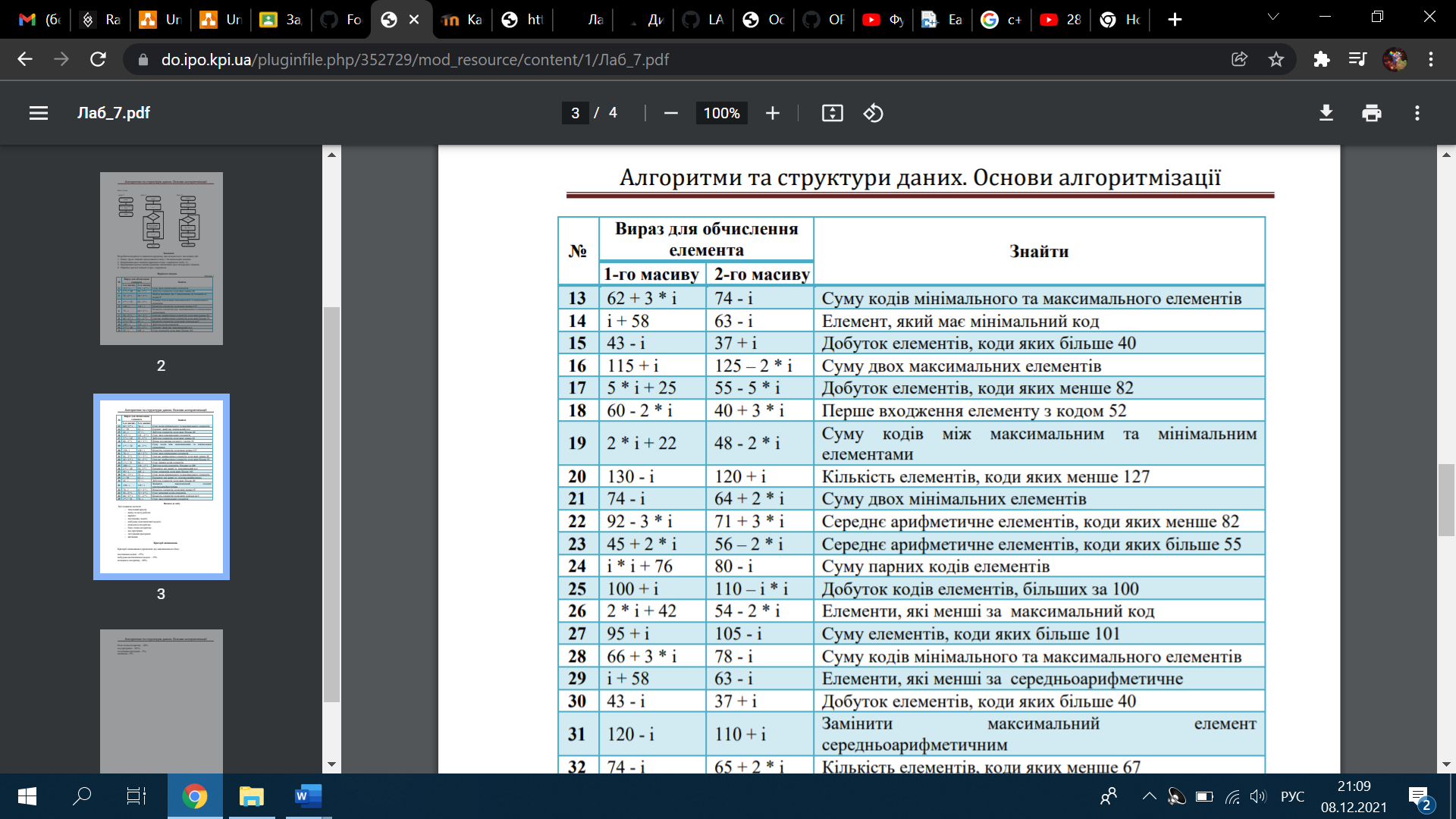
**Завдання**:

Розробити алгоритм та написати програму, яка складається з наступних дій:

1. Опису трьох змінних індексованого типу з 10 символьних значень.

2. Ініціювання двох змінних виразами згідно з варіантом.

3. Ініціювання третьої змінної рівними значеннями двох попередніх змінних. 4. Обробки третьої змінної згідно з варіантом



**Розв’язок:**

1)Постановка задачі. Результатом розв’язку буде кількість елементів, коди яких менше 127. Треба знайти кількість елементів, коди яких менше 127 за допомогою лінійного пошуку в послідовностях та альтернативної форми вибору. Спочатку треба описати три змінні індексованого типу з 10 символьних значень. Далі ініціювати дві змінні виразами згідно з варіантом(1-ий масив: 130-і, 2-ий масив:120+і).Далі ініціювати третю змінну рівними значеннями двох попередніх змінних.

**Побудова математичної моделі:**

|  |  |  |  |
| --- | --- | --- | --- |
| Змінна | Тип | Ім’я | Призначення |
| Перший масив | Символьний | arr1 | Проміжні дані |
| Другий масив | Символьний | arr2 | Проміжні дані |
| Третій масив | Символьний | arr3 | Проміжні дані |
| Лічильник і | Цілий | i | Проміжні дані |
| Лічильник j | Цілий | j | Проміжні дані |
| Кількість | Ціллий | quant | Результат |
| Процедура viraz1 | - | viraz1 | Проміжні дані |
| Процедура viraz2 | - | viraz2 | Проміжні дані |
| Процедура viraz3 | - | viraz3 | Проміжні дані |
| Процедура vivod | - | vivod | Проміжні дані |
| Функція final | Цілий | final | Проміжні дані |

Перший та другий масиви(розмірність яких - 10) задаються елементами з кодами: 130-і та 120+і, відповідно. В цьому випадку і – лічильник, який набуває значень від 1 до 9 включно ([1;9]). За умовою задачі третій масив – це спільні елементи першого та другого масивів. За допомогою таблиці можемо зрозуміти, які саме символи входять до наших масивів. До першого:[ В, Б, А, ⌂, ~, }, |, {, z, y] До другого:[ x, y, z, {, |, }, ~, ⌂, А, Б] До третього:[ Б, А, ⌂, ~, }, |, {, z, y, 0], в масиві з’являється нуль, бо порожні комірки масива ми заповняємо нулями, щоб не створювати динамічний масив. І потім знаходиться кількість цих елементів, ASCII код строго менший 127, за допомогою змінної quant.

Крок 1. Визначимо основні дії.

Крок 2. Деталізуємо дію ініціювання першого та другого масивів за допомогою процедури viraz1 та viraz2.

Крок 3. Деталізуємо дію знаходження третього масиву за допомогою процедури viraz3.

Крок 4. Деталізуємо дію виведення масива за допомогою процедури vivod.

Крок 5. Деталізуємо дію знаходження кількості елементів третього масиву, код яких строго менше 127 за допомогою функції final.

Крок 6. Деталізуємо дію знаходження кількості елементів третього масиву, код яких строго менше 127 за допомогою підпрограм в основному тілі програми.

**Псевдокод:**

**Крок 1**

**Початок**

i,j,quant

arr1[10], arr2[10], arr3[10]{}

ініціюємо перший та другий масиви за допомогою процедур viarz1 та viraz2.

знаходимо третій масив за допомогою процедури viraz3.

виведення масивів за допомогою процедури vivod.

знаходимо кількість елементів третього масиву, код яких строго менше 127 за допомогою функції final

виведення quant

**Кінець**

**Процедура**

**viraz1(\*arr1):**

Деталізуємо дію ініціювання першого масиву за допомогою процедури viraz1.

**Все процедура**

**Процедура**

**viraz2(\*arr2):**

Деталізуємо дію ініціювання першого масиву за допомогою процедури viraz2.

**Все процедура**

**Процедура**

**viraz3(\*arr3):**

Деталізуємо дію ініціювання першого масиву за допомогою процедури viraz3.

**Все процедура**

**Процедура**

**vivod(\*arr):**

Деталізуємо дію ініціювання першого масиву за допомогою процедури viraz3.

**Все процедура**

**Функція**

**final(\*arr3):**

Деталізуємо дію знаходження кількості елементів третього масиву, код яких строго менше 127 за допомогою функції quant.

**Все функція**

**Крок 2**

**Початок**

i,j,quant

arr1[10], arr2[10], arr3[10]{}

ініціюємо перший та другий масиви за допомогою процедур viarz1 та viraz2.

знаходимо третій масив за допомогою процедури viraz3.

виведення масивів за допомогою процедури vivod.

знаходимо кількість елементів третього масиву, код яких строго менше 127 за допомогою функції final

виведення quant

**Кінець**

**Процедура**

**viraz1(\*arr1):**

**повторити для і від 0 до 10 не включно з кроком 1:**

\*arr1 = 130 - i

arr1++

**все повторити**

**Все процедура**

**Процедура**

**viraz2(\*arr2):**

**повторити для і від 0 до 10 не включно з кроком 1:**

\*arr2 = 120 + i

аrr2++

**все повторити**

**Все процедура**

**Процедура**

**viraz3(\*arr3):**

Деталізуємо дію ініціювання першого масиву за допомогою процедури viraz3.

**Все процедура**

**Процедура**

**vivod(\*arr):**

Деталізуємо дію ініціювання першого масиву за допомогою процедури viraz3.

**Все процедура**

**Функція**

**final(\*arr3):**

Деталізуємо дію знаходження кількості елементів третього масиву, код яких строго менше 127 за допомогою функції quant.

**Все функція**

**Крок 3**

**Початок**

i,j,quant

arr1[10], arr2[10], arr3[10]{}

ініціюємо перший та другий масиви за допомогою процедур viarz1 та viraz2.

знаходимо третій масив за допомогою процедури viraz3.

виведення масивів за допомогою процедури vivod.

знаходимо кількість елементів третього масиву, код яких строго менше 127 за допомогою функції final

виведення quant

**Кінець**

**Процедура**

**viraz1(\*arr1):**

**повторити для і від 0 до 10 не включно з кроком 1:**

\*arr1 = 130 - i

arr1++

**все повторити**

**Все процедура**

**Процедура**

**viraz2(\*arr2):**

**повторити для і від 0 до 10 не включно з кроком 1:**

\*arr2 = 120 + i

аrr2++

**все повторити**

**Все процедура**

**Процедура**

**viraz3(\*arr3):**

**повторити для і від 0 до 10 не включно з кроком 1:**

**повторити для j від 0 до 10 не включно з кроком 1:**

**якщо** (\*arr1==\*arr2) то \*arr3 = \*arr2

arr3++

**все якщо**

**якщо** (\*arr3==0)

**то**

\*arr3 = 48

**все якщо**

arr2++

**все повторити**

arr1++

arr2-=10

**все повторити**

**Все процедура**

**Процедура**

**vivod(\*arr):**

Деталізуємо дію ініціювання першого масиву за допомогою процедури viraz3.

**Все процедура**

**Функція**

**final(\*arr3):**

Деталізуємо дію знаходження кількості елементів третього масиву, код яких строго менше 127 за допомогою функції quant.

**Все функція**

**Крок 4**

**Початок**

i,j,quant

arr1[10], arr2[10], arr3[10]{}

ініціюємо перший та другий масиви за допомогою процедур viarz1 та viraz2.

знаходимо третій масив за допомогою процедури viraz3.

виведення масивів за допомогою процедури vivod.

знаходимо кількість елементів третього масиву, код яких строго менше 127 за допомогою функції final

виведення quant

**Кінець**

**Процедура**

**viraz1(\*arr1):**

**повторити для і від 0 до 10 не включно з кроком 1:**

\*arr1 = 130 - i

arr1++

**все повторити**

**Все процедура**

**Процедура**

**viraz2(\*arr2):**

**повторити для і від 0 до 10 не включно з кроком 1:**

\*arr2 = 120 + i

аrr2++

**все повторити**

**Все процедура**

**Процедура**

**viraz3(\*arr3):**

**повторити для і від 0 до 10 не включно з кроком 1:**

**повторити для j від 0 до 10 не включно з кроком 1:**

**якщо** (\*arr1==\*arr2) то \*arr3 = \*arr2

arr3++

**все якщо**

**якщо** (\*arr3==0)

**то**

\*arr3 = 48

**все якщо**

arr2++

**все повторити**

arr1++

arr2-=10

**все повторити**

**Все процедура**

**Процедура**

**vivod(\*arr):**

**повторити для і від 0 до 10 не включно з кроком 1:**

виведення \*(arr+i)

**все повторити**

**Все процедура**

**Функція**

**final(\*arr3):**

Деталізуємо дію знаходження кількості елементів третього масиву, код яких строго менше 127 за допомогою функції quant.

**Все функція**

**Крок 5**

**Початок**

i,j,quant

arr1[10], arr2[10], arr3[10]{}

ініціюємо перший та другий масиви за допомогою процедур viarz1 та viraz2.

знаходимо третій масив за допомогою процедури viraz3.

виведення масивів за допомогою процедури vivod.

знаходимо кількість елементів третього масиву, код яких строго менше 127 за допомогою функції final

виведення quant

**Кінець**

**Процедура**

**viraz1(\*arr1):**

**повторити для і від 0 до 10 не включно з кроком 1:**

\*arr1 = 130 - i

arr1++

**все повторити**

**Все процедура**

**Процедура**

**viraz2(\*arr2):**

**повторити для і від 0 до 10 не включно з кроком 1:**

\*arr2 = 120 + i

аrr2++

**все повторити**

**Все процедура**

**Процедура**

**viraz3(\*arr3):**

**повторити для і від 0 до 10 не включно з кроком 1:**

**повторити для j від 0 до 10 не включно з кроком 1:**

**якщо** (\*arr1==\*arr2) то \*arr3 = \*arr2

arr3++

**все якщо**

**якщо** (\*arr3==0)

**то**

\*arr3 = 48

**все якщо**

arr2++

**все повторити**

arr1++

arr2-=10

**все повторити**

**Все процедура**

**Процедура**

**vivod(\*arr):**

**повторити для і від 0 до 10 не включно з кроком 1:**

виведення \*(arr+i)

**все повторити**

**Все процедура**

**Функція**

**final(\*arr3):**

quant = 0

**повторити для і від 0 до 10 не включно з кроком 1:**

**якщо**

(\*arr\_3 < 127 && \*arr\_3 != 48) то number+=1

arr\_3++

**все якщо**

**все повторити**

**Все функція**

**Крок 6**

**Початок**

i,j,quant

viraz1(arr1)

viraz2(arr2)

viraz3(arr1, arr2, arr3);

vivod(arr1)

vivod(arr2)

vivod(arr3)

quant = final(arr3)

виведення quant

**Кінець**

**Процедура**

**viraz1(\*arr1):**

**повторити для і від 0 до 10 не включно з кроком 1:**

\*arr1 = 130 - i

arr1++

**все повторити**

**Все процедура**

**Процедура**

**viraz2(\*arr2):**

**повторити для і від 0 до 10 не включно з кроком 1:**

\*arr2 = 120 + i

аrr2++

**все повторити**

**Все процедура**

**Процедура**

**viraz3(\*arr3):**

**повторити для і від 0 до 10 не включно з кроком 1:**

**повторити для j від 0 до 10 не включно з кроком 1:**

**якщо** (\*arr1==\*arr2) то \*arr3 = \*arr2

arr3++

**все якщо**

**якщо** (\*arr3==0)

**то**

\*arr3 = 48

**все якщо**

arr2++

**все повторити**

arr1++

arr2-=10

**все повторити**

**Все процедура**

**Процедура**

**vivod(\*arr):**

**повторити для і від 0 до 10 не включно з кроком 1:**

виведення \*(arr+i)

**все повторити**

**Все процедура**

**Функція**

**final(\*arr3):**

quant = 0

**повторити для і від 0 до 10 не включно з кроком 1:**

**якщо**

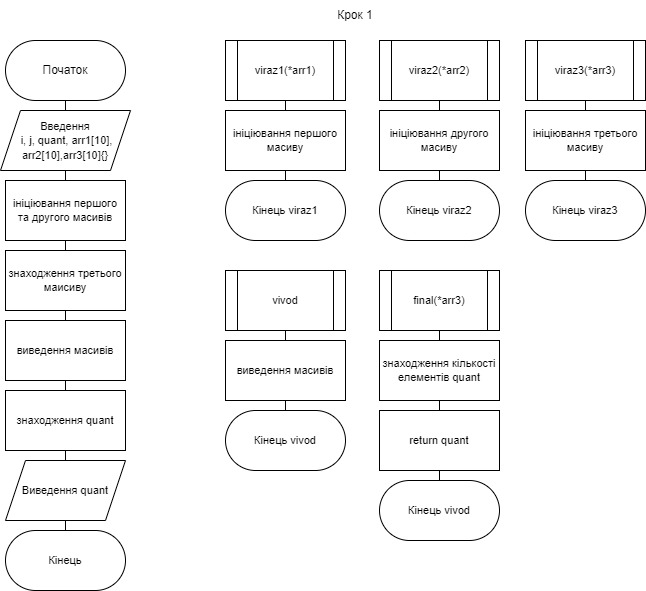
(\*arr\_3 < 127 && \*arr\_3 != 48) то number+=1

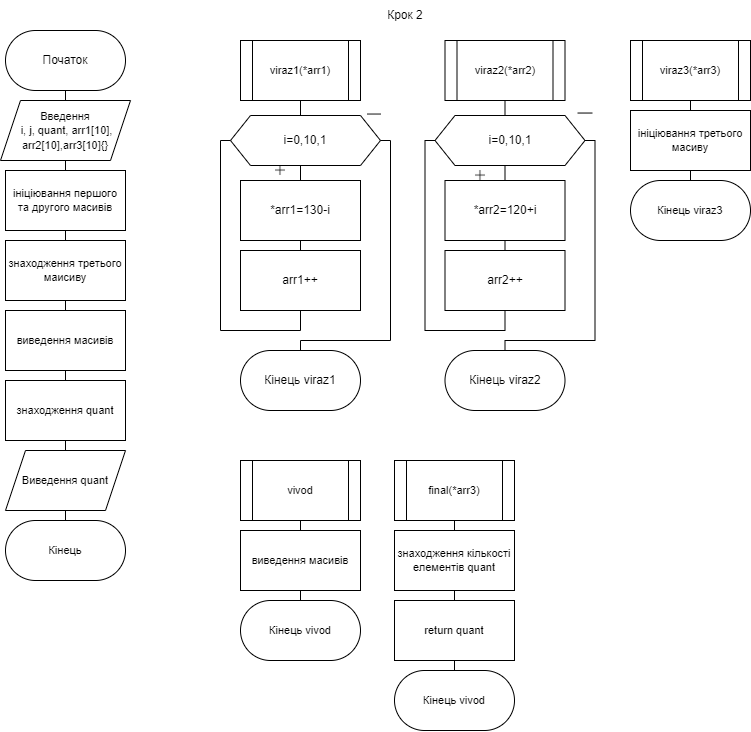
arr\_3++

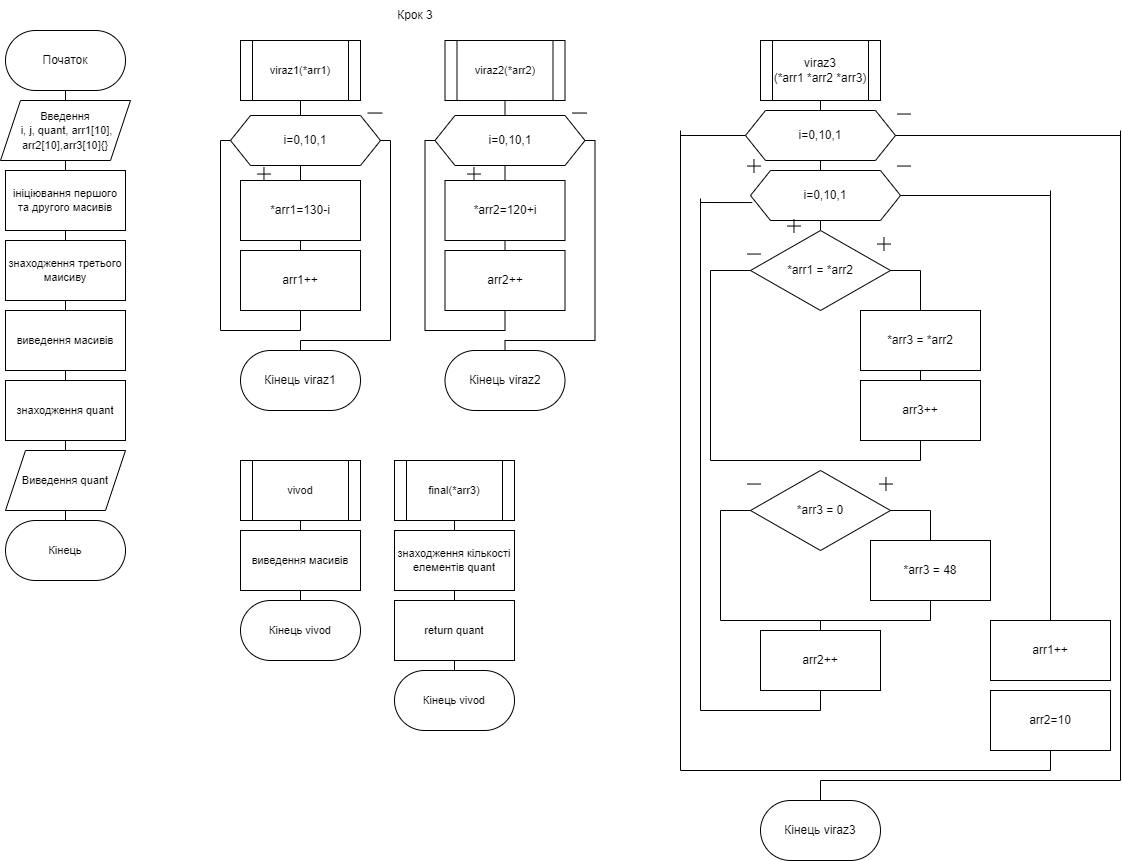
**все якщо**

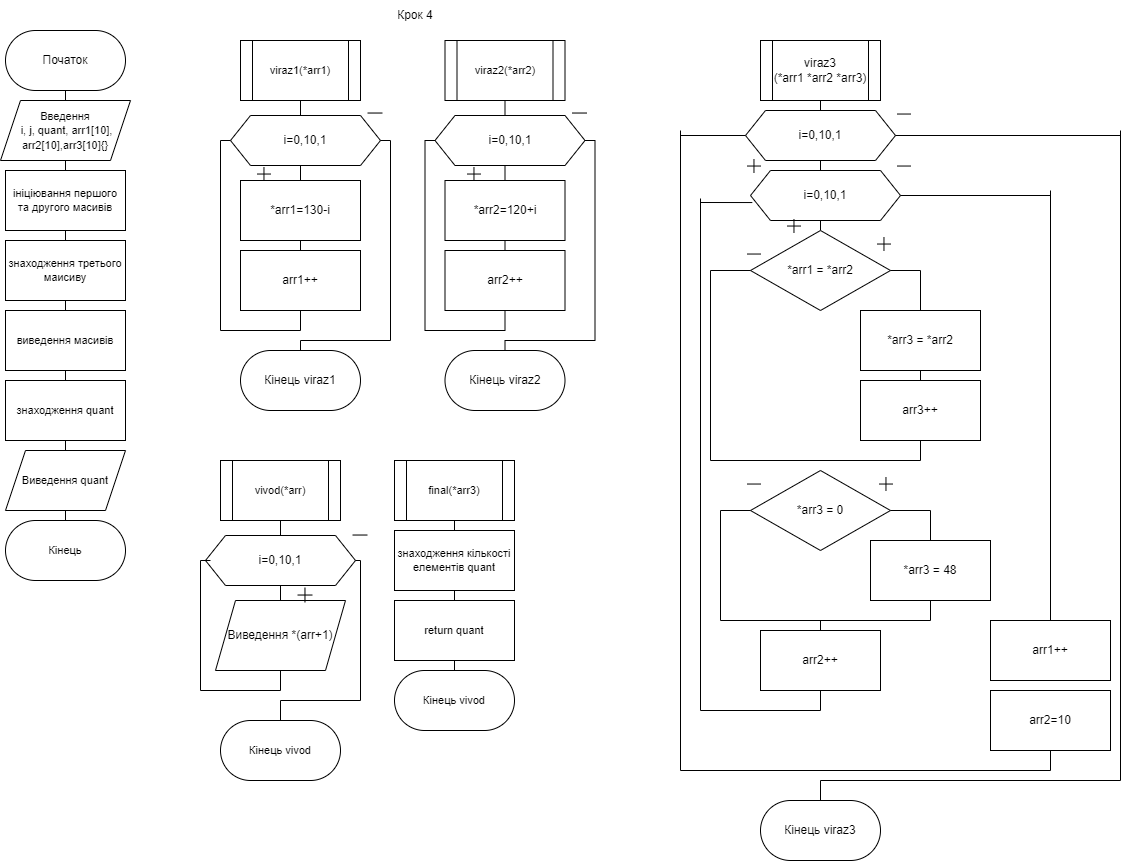
**все повторити**

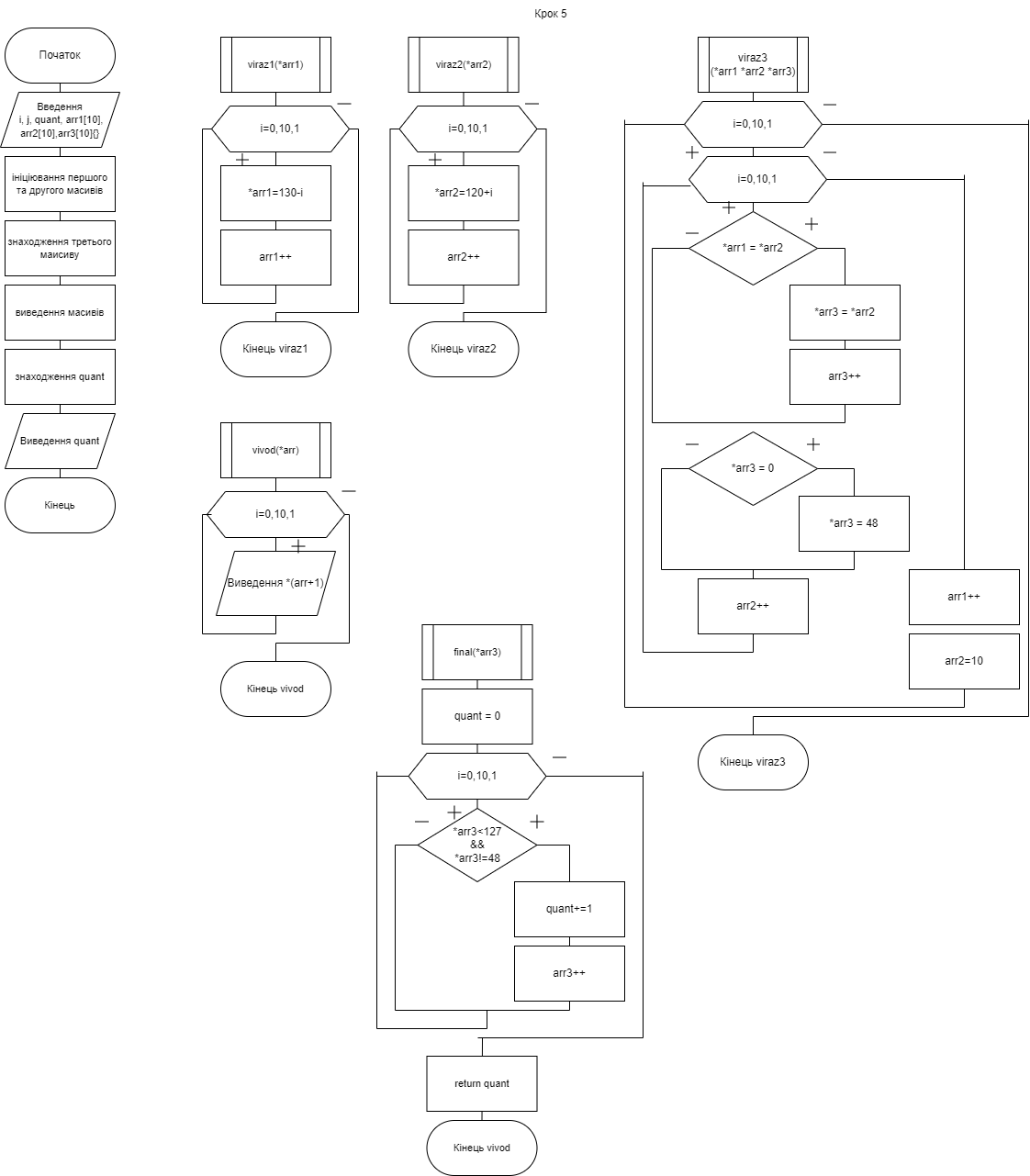
**Все функція**

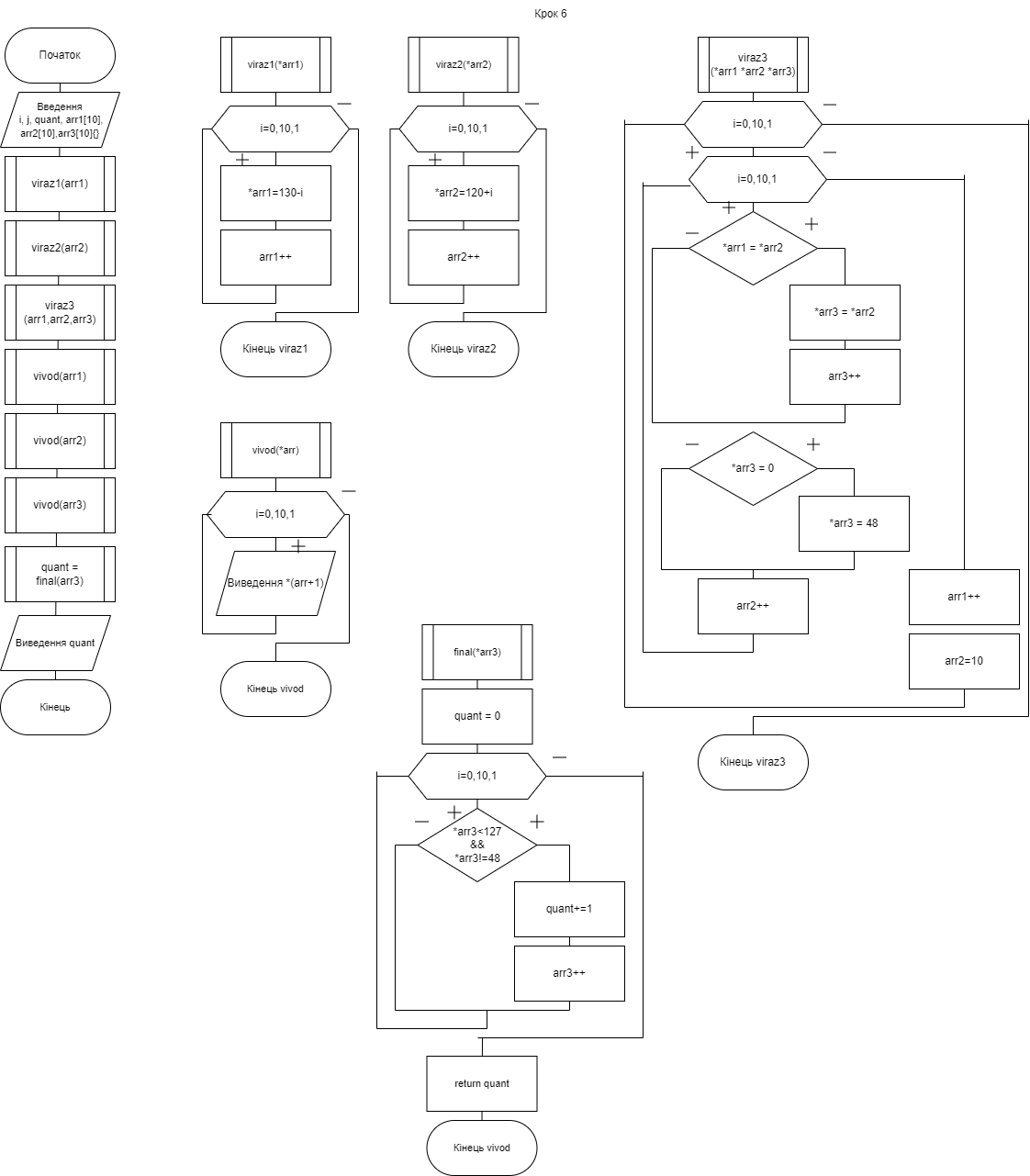
****

****

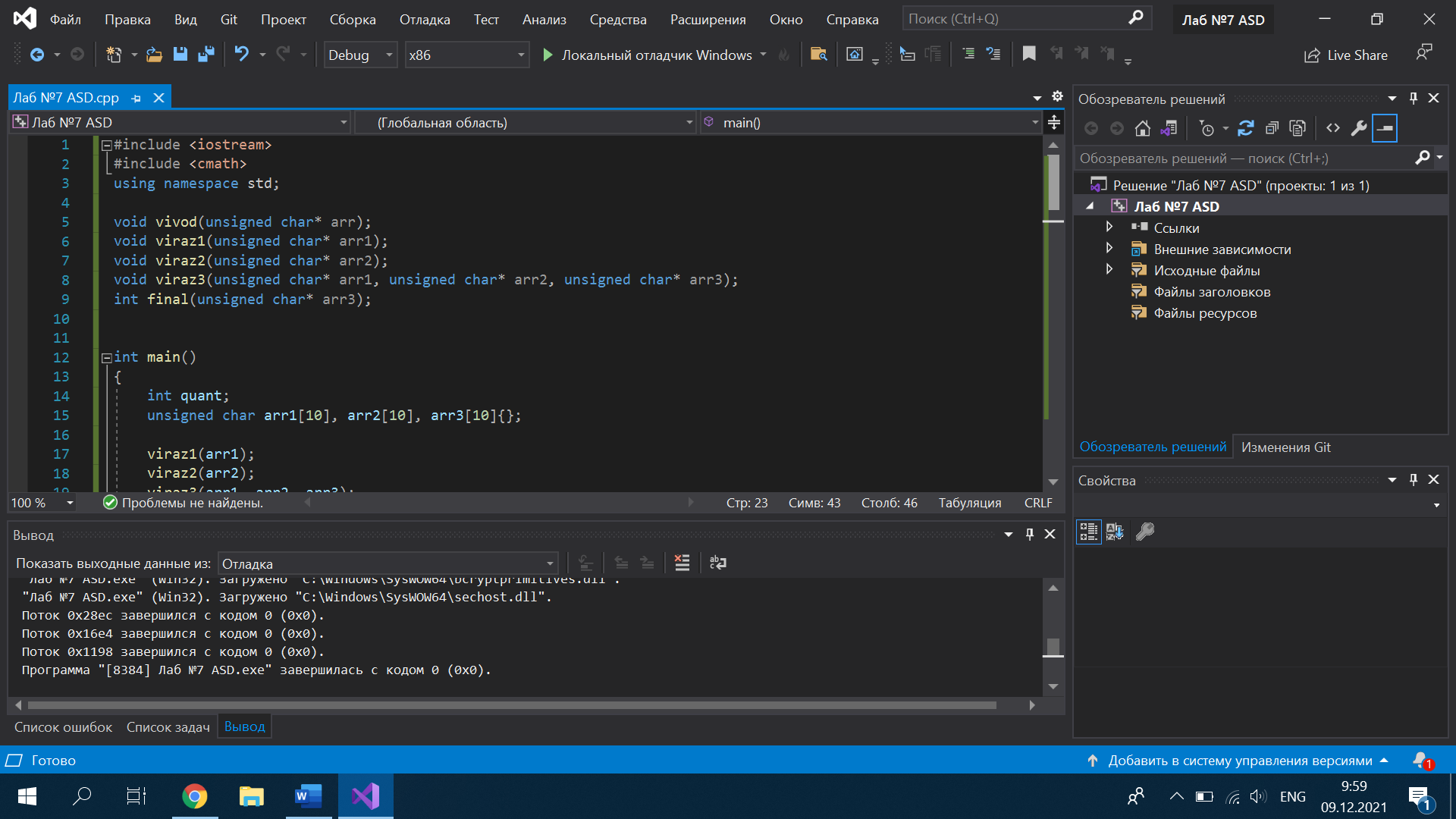


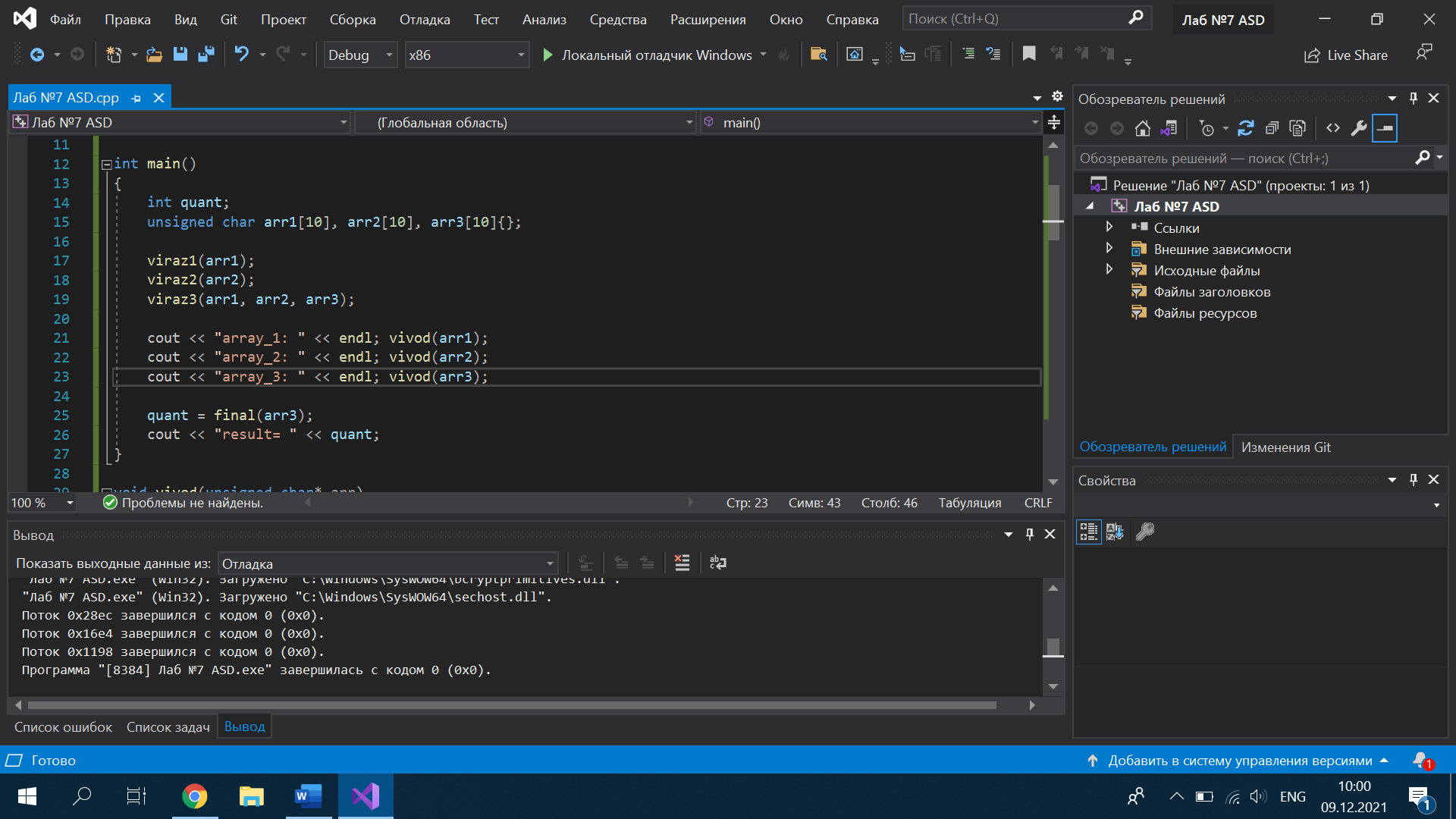


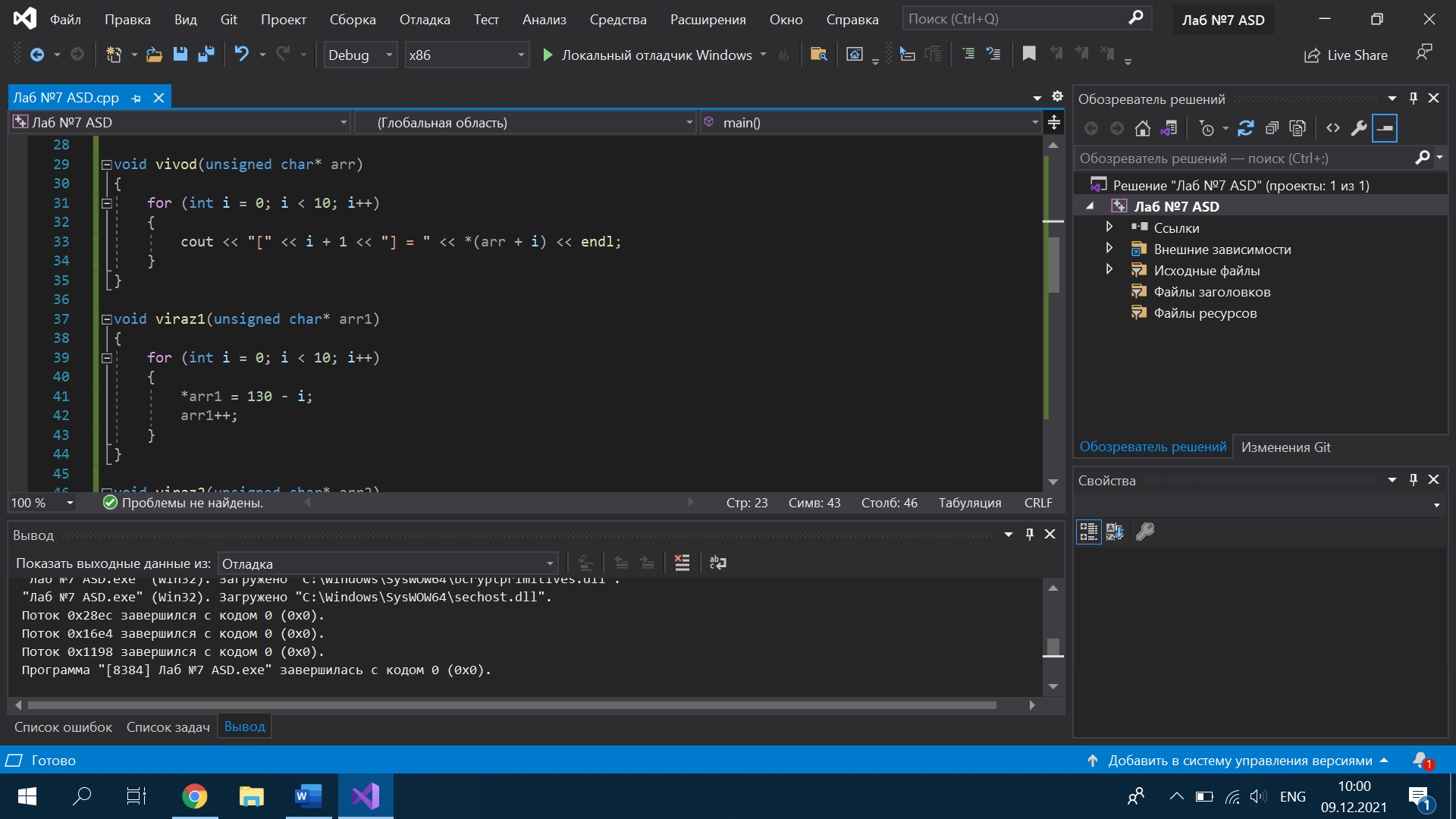


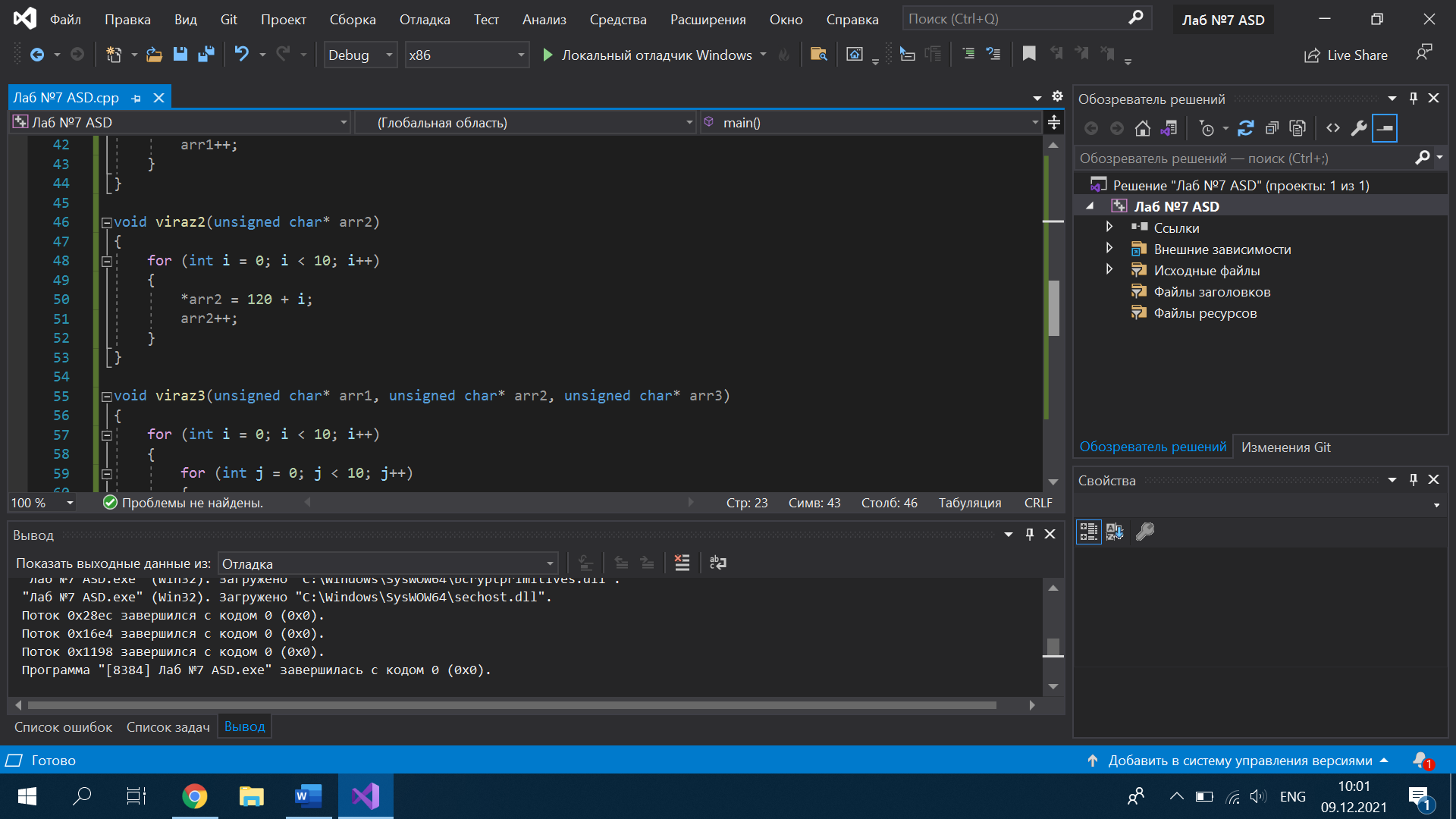


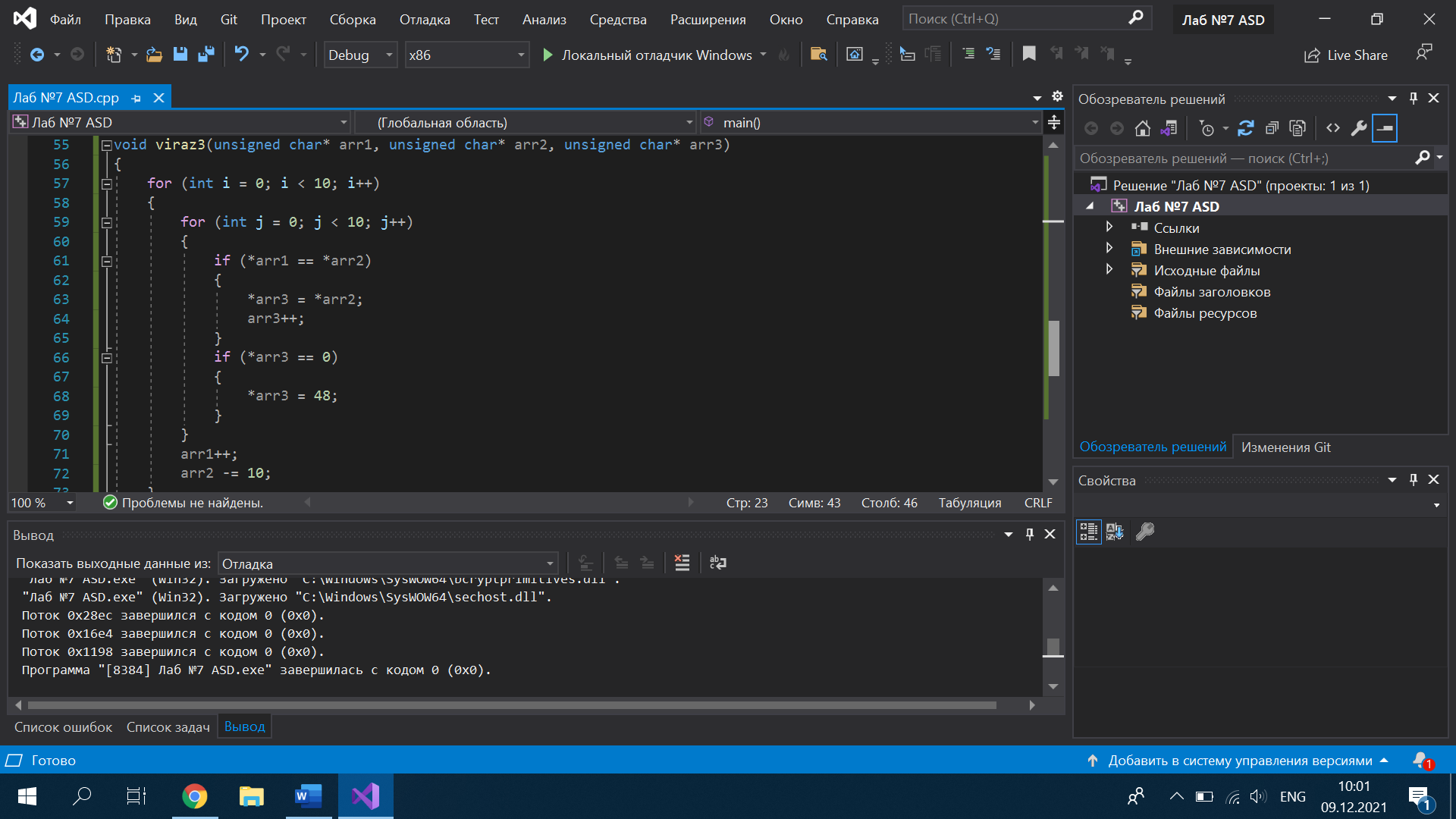
**Код на С++:**

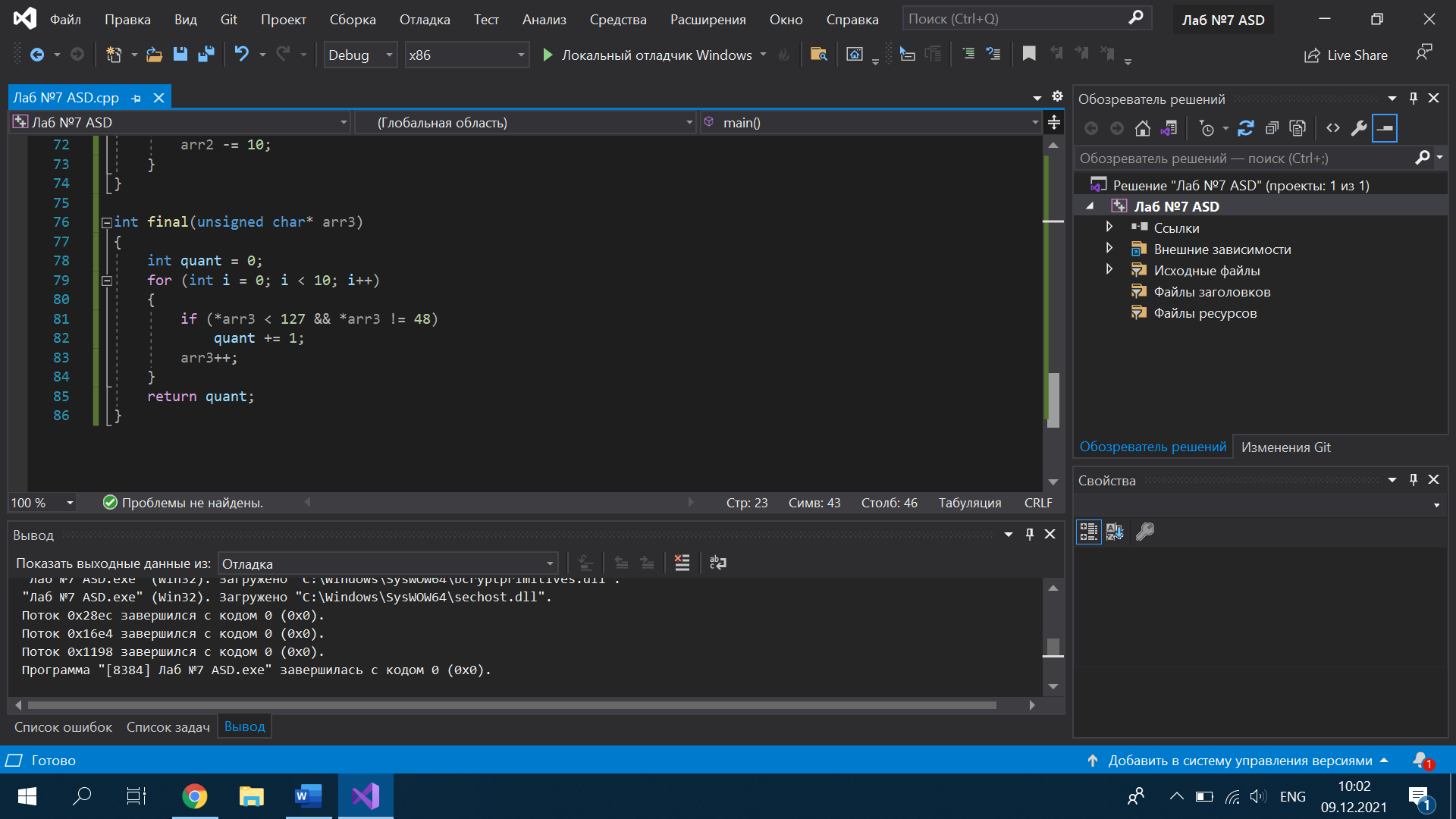


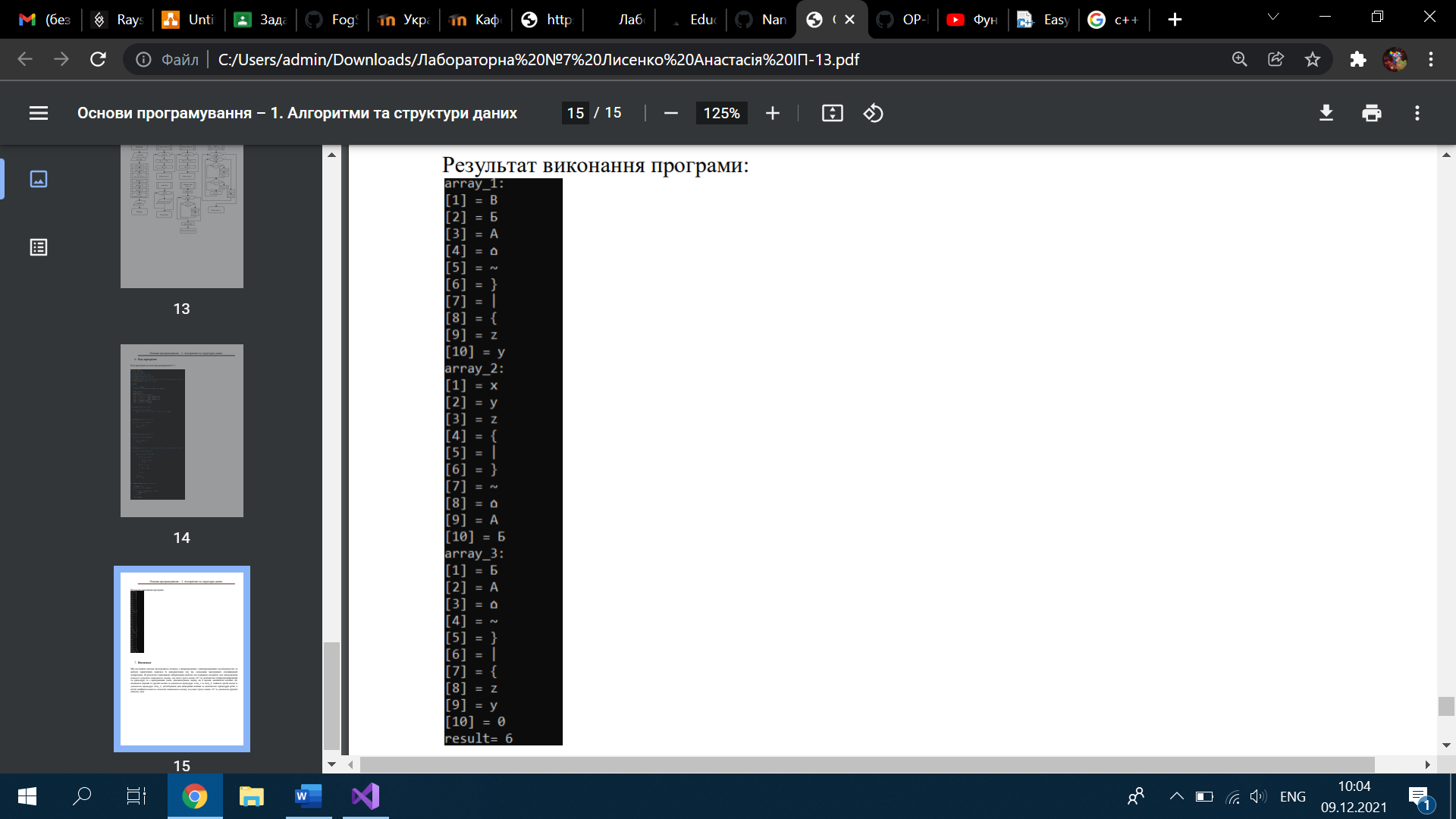












**Висновок:** Під час даної лабораторної роботи були досліджені методи послідовного пошуку у невпорядкованих послідовностях. Основними способами організації такого алгоритму стало використання альтернативної форми вибору та оператора повторення. Були набуті навички використання їх у програмних специфікаціях. Були побудовані блок схема та математичка модель. Для обчислення виразу була створена елементарна програма. Результати програми виявилися правильними, що стверджує на її дієвість. Завдання було виконано на мові програмування: C++.