



**«Московский государственный технический университет
имени Н.Э. Баумана»
(национальный исследовательский университет)
(МГТУ им. Н.Э. Баумана)**

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ
КАФЕДРА КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)

Отчет

по лабораторной работе № 1

Название лабораторной работы: Консольные приложения Visual C++ в среде Microsoft Studio 2022 Community

Дисциплина: Алгоритмизация и программирование

Студент гр. ИУ6-14Б

20.09.2023

А.С. Воеводин

(Подпись, дата)

(И.О. Фамилия)

Преподаватель

20.09.2023

О.А. Веселовская

(Подпись, дата)

(И.О. Фамилия)

Цель работы – закрепление теоретических знаний и формирование практических навыков, необходимых для разработки программ на языках программирования С и С++, формирование умений создания, тестирования и отладки созданных программ.

Задание – познакомиться с работой консольных приложений языка С++.

Ход работы:

- Установка Visual Studio
- Создание нового проекта
- Написание алгоритма Евклида
- Выполнение отладки программы
- Просмотр вывода ошибок
- Изображение схемы алгоритма Евклида Google Drawings
- Изображение схемы алгоритма Евклида Draw.io

Для начала запустим среду разработки Visual Studio без открытого проекта:

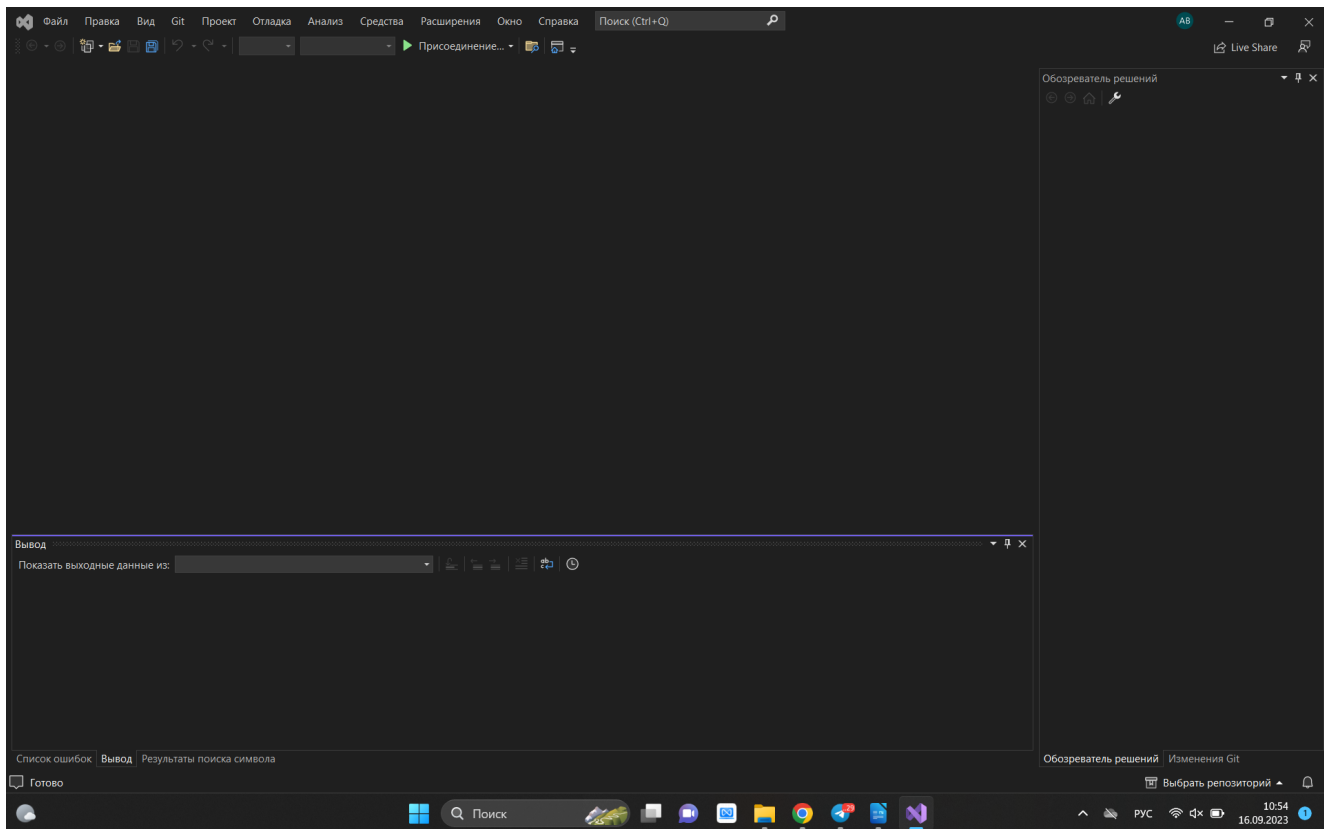


Рисунок 1 – Вид главного окна среды при отсутствии открытого проекта

Далее создадим пустой проект под названием `lab1_1`. Для этого в меню слева сверху необходимо выполнить следующую последовательность действий:

Файл=>Создать=>Проект... Получим следующее:

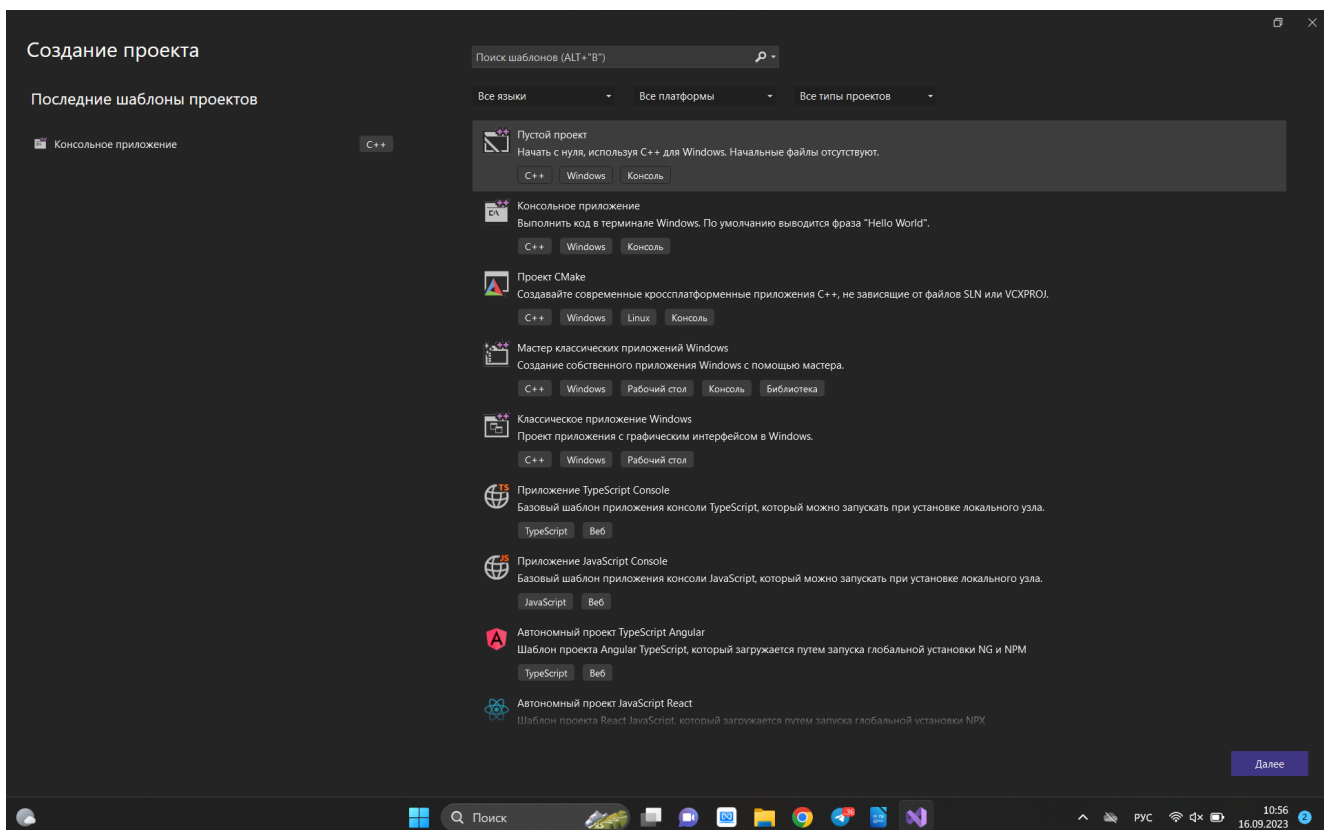


Рисунок 2 – Вид окна выбора типа проекта

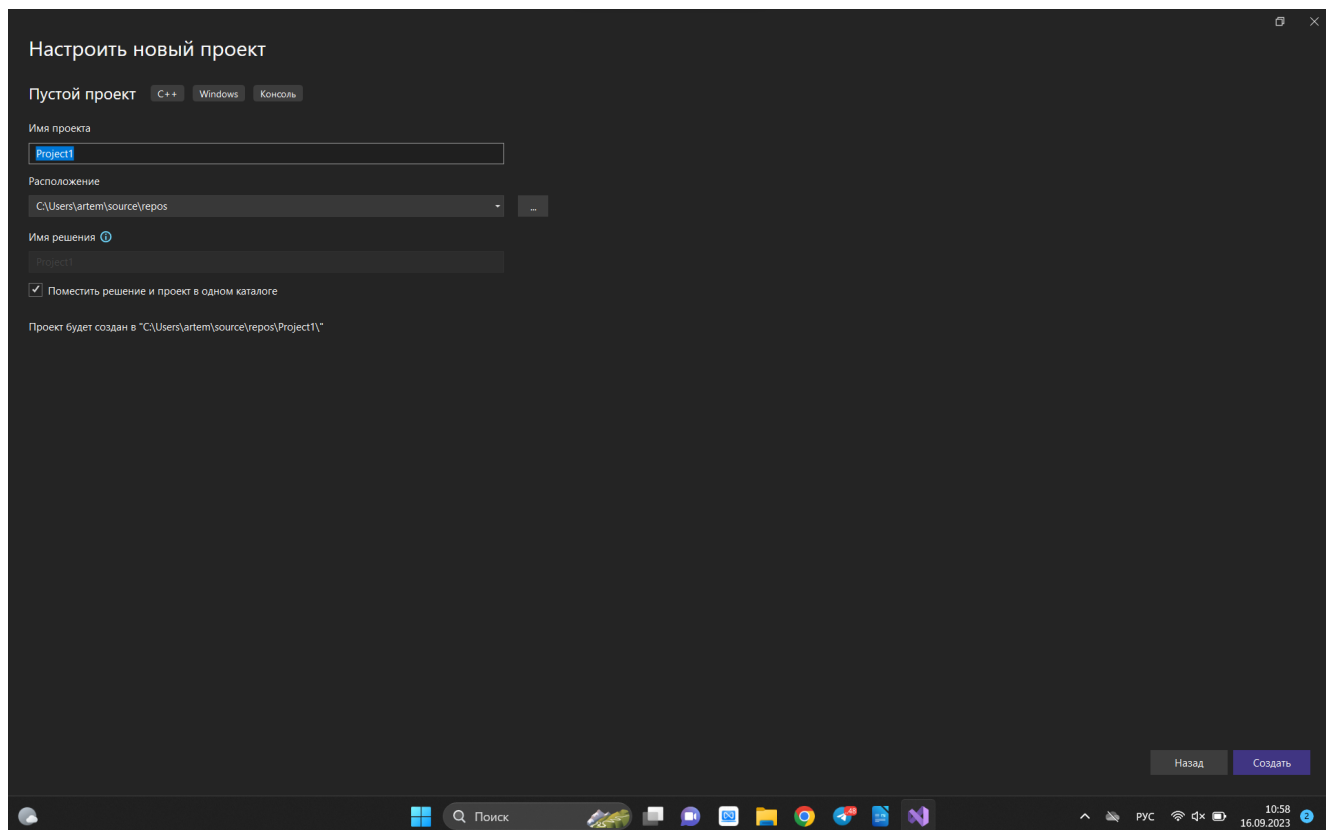


Рисунок 3 – Вид окна выбора имени и местоположения проекта

После этого создадим файл Source.cpp для дальнейшей работы:

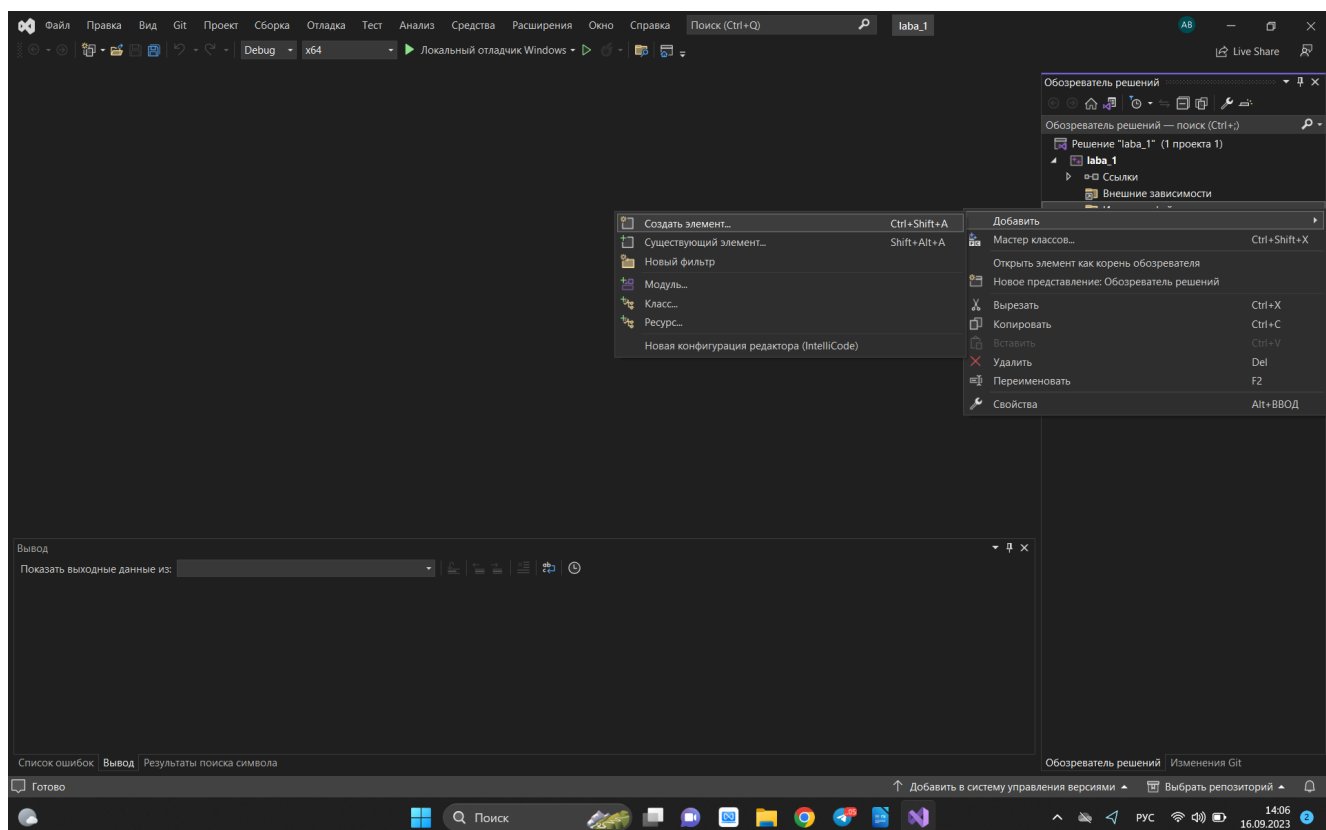


Рисунок 4 – Добавление файлов к проекту

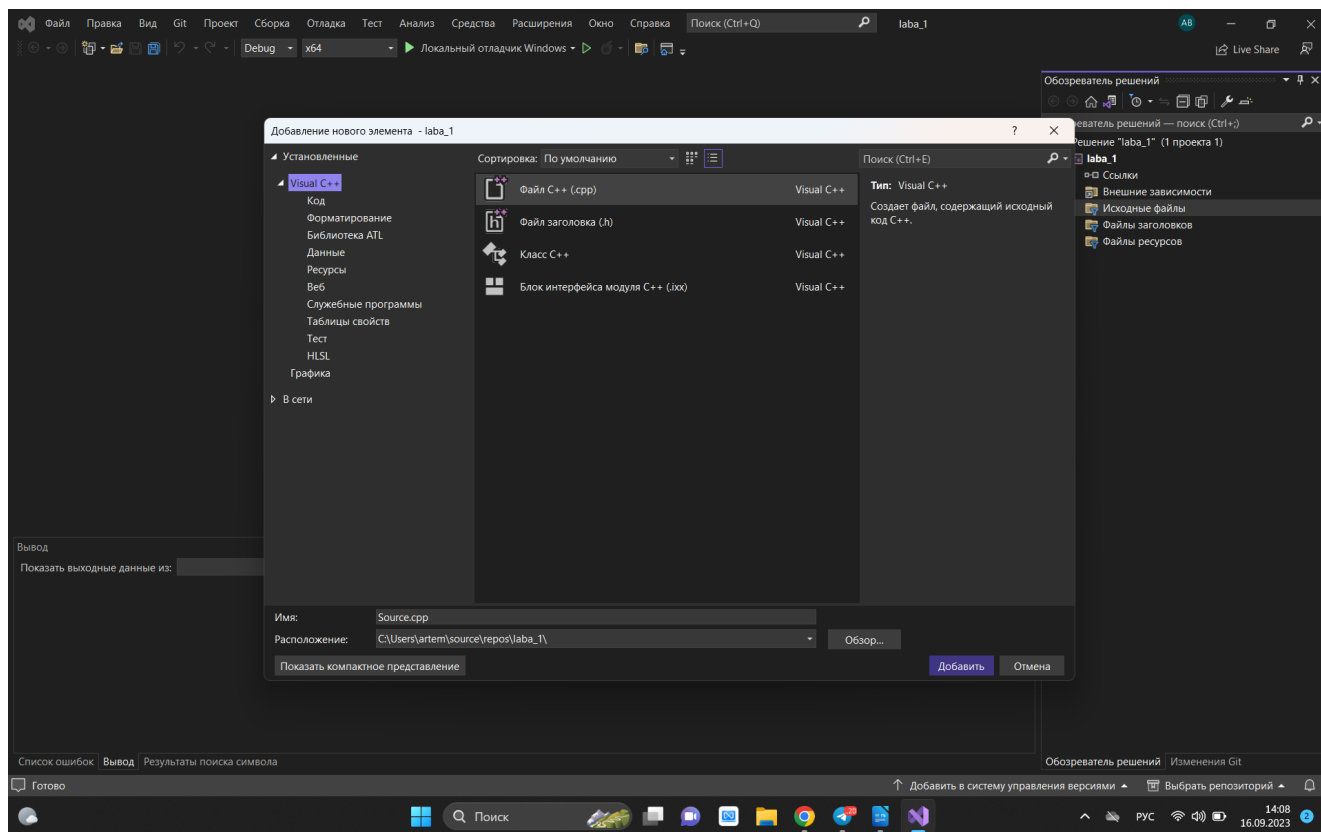


Рисунок 5 – Выбор типа добавляемого файла

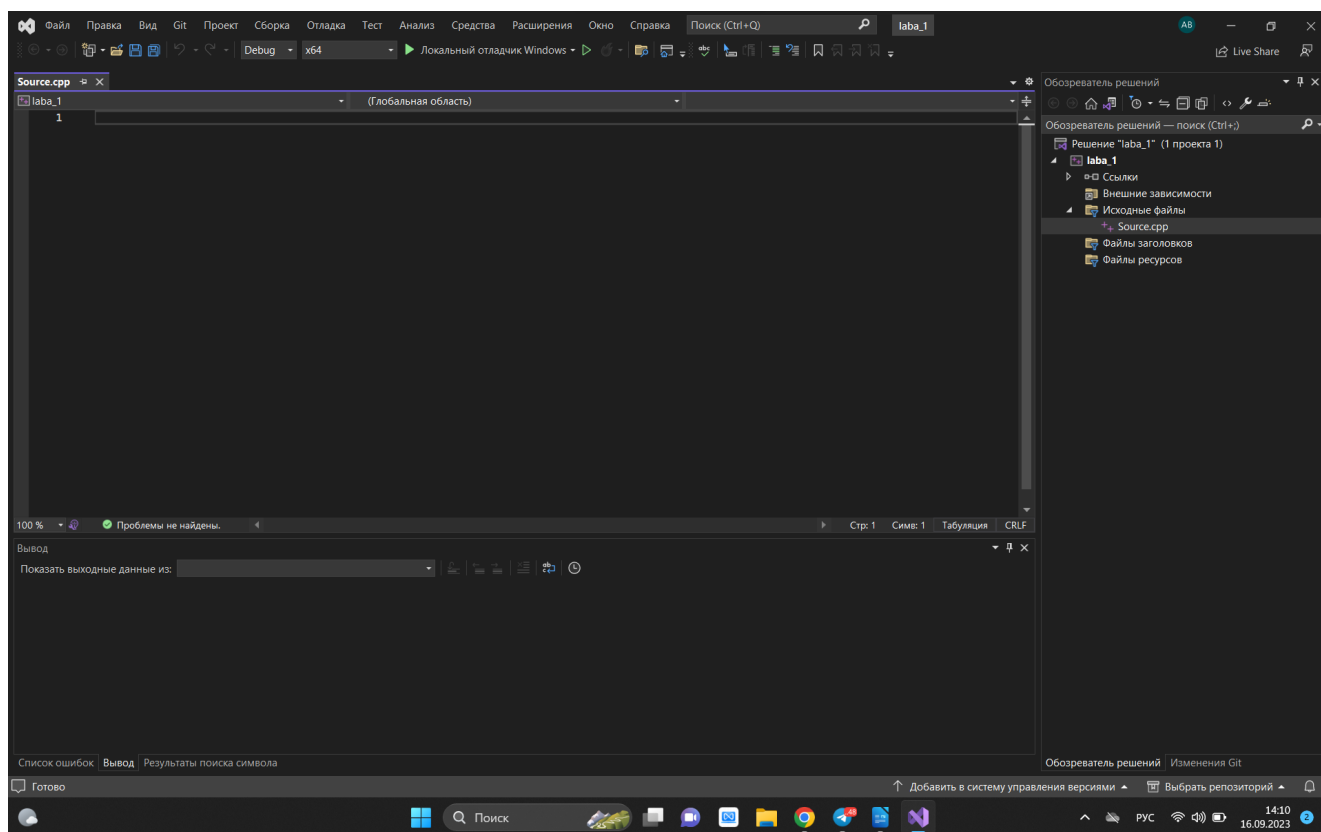


Рисунок 6 – Переход в окно текстового редактора для ввода программы

Напишем простейшую программу, выводящая на консоль фразу “Hello World!”:

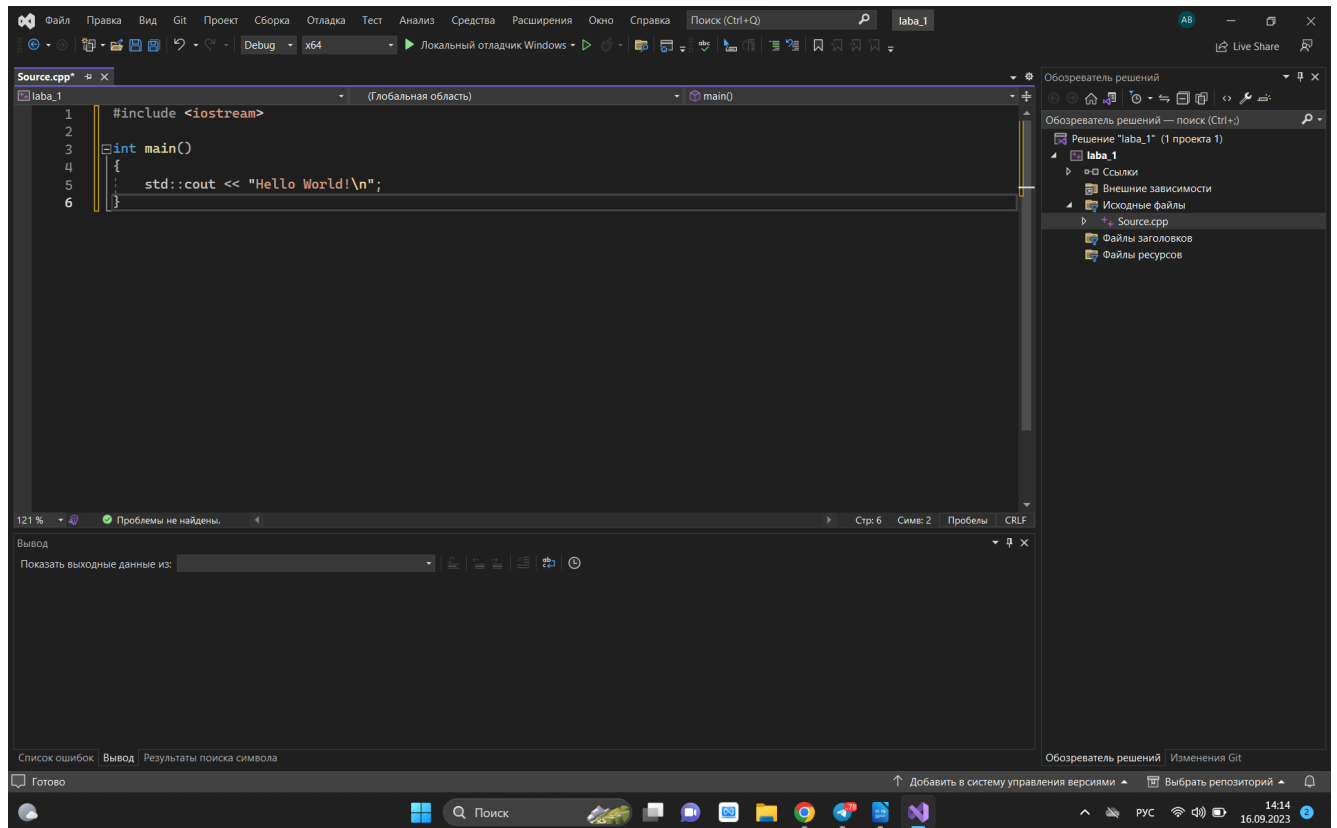


Рисунок 7 – Программа, выводящая на консоль фразу “Hello World!”

Далее напишем более сложную программу, а именно – нахождение наибольшего общего делителя двух целых чисел:

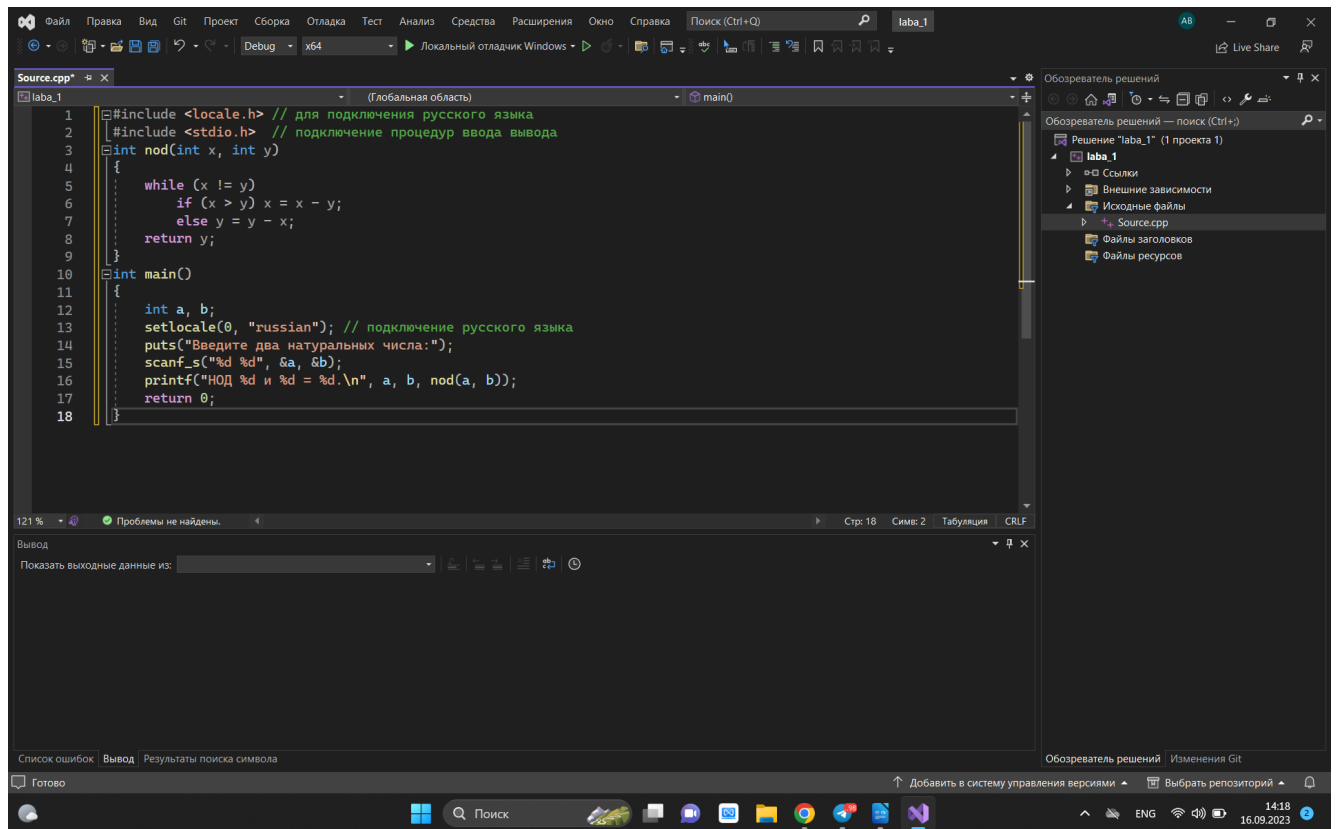


Рисунок 8 – Программа для нахождения наибольшего общего делителя двух целых чисел

В случае, если мы допустили ошибку, компилятор не даст собрать проект.

Сигналом этого послужит всплывающее окно об ошибке сборки.

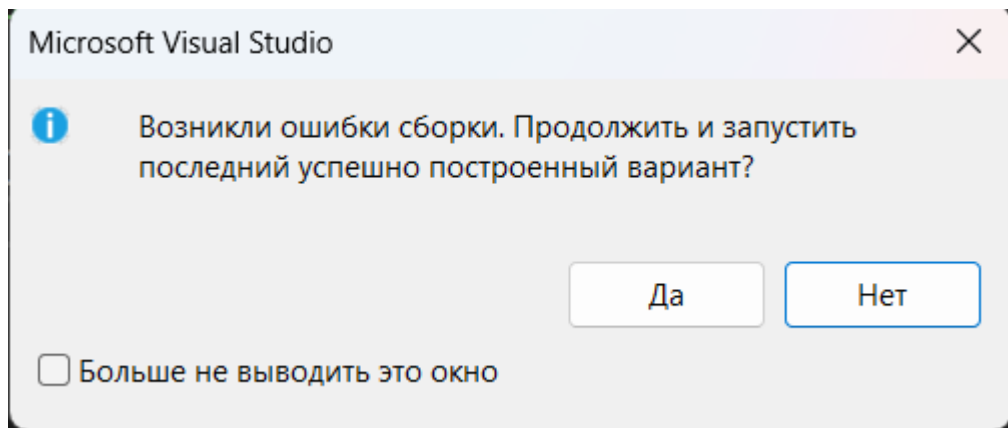


Рисунок 9 – Запрос на выполнение программы при наличии ошибок

Если же ошибок нет, то на консоль выведется следующее:

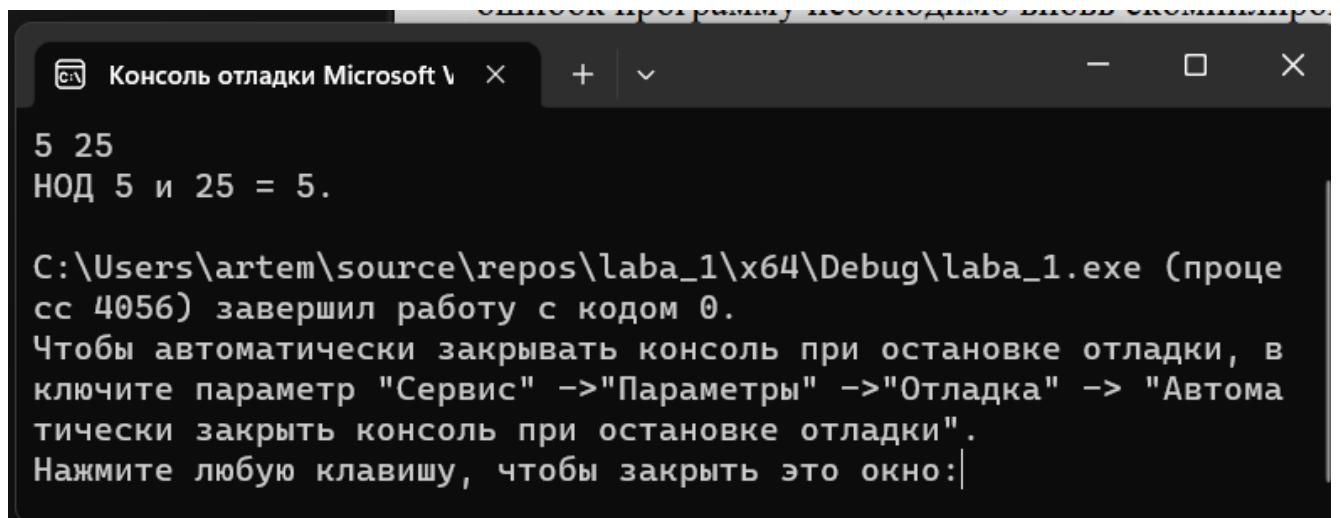


Рисунок 10 – Окно результата

Чтобы упростить структуру программы, создадим в разделе **Файлы заголовков** 2 файла – Nod.h и Nod.cpp, и скопируем код функции Nod в файл Nod.cpp:

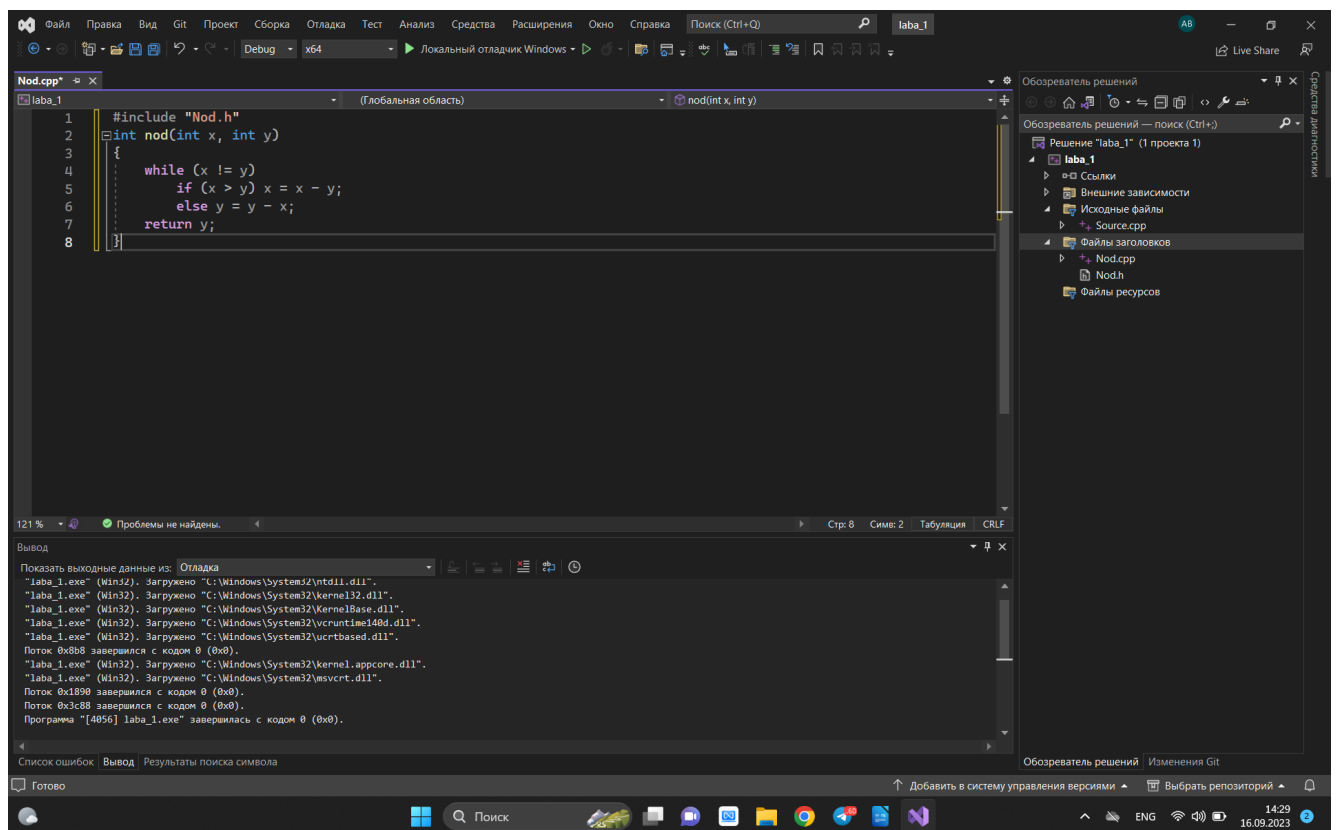


Рисунок 11 – Файл реализации Nod.cpp

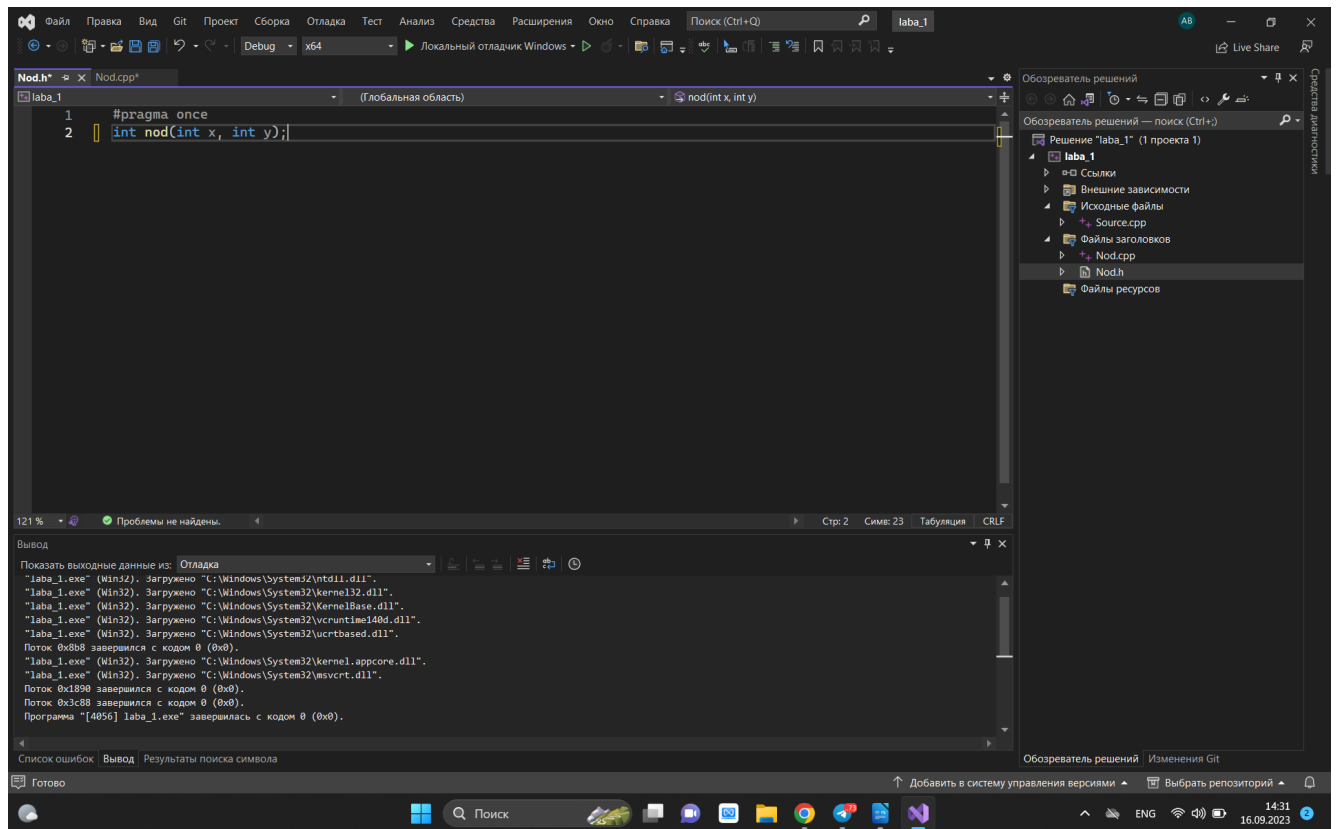


Рисунок 12 – Заголовочный файл Nod.h

Подключим получившуюся функцию в исходном файле:

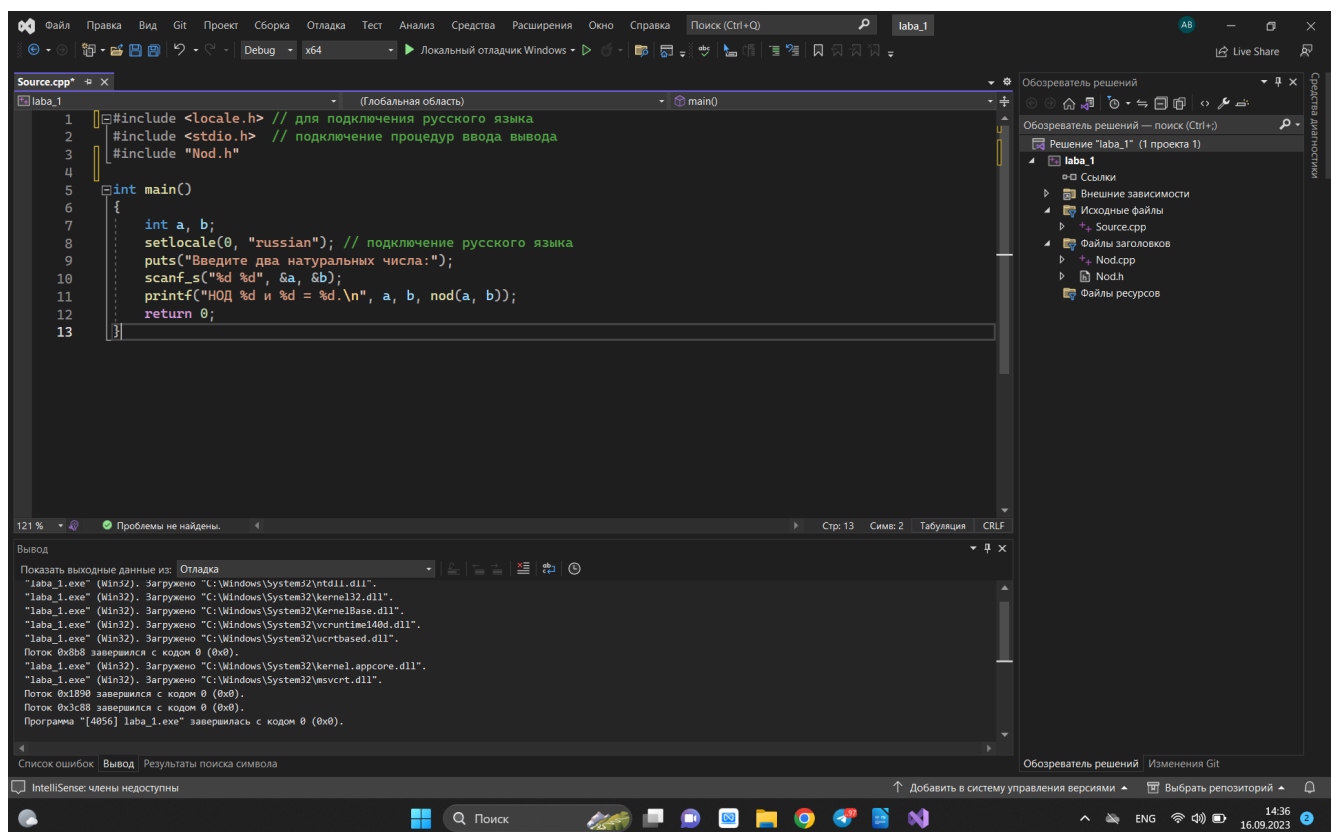


Рисунок 13 – Подключение заголовочного файла в файле Source.cpp

Если требуется запустить программу в режиме отладки, то нажимаем на **Отладка**:

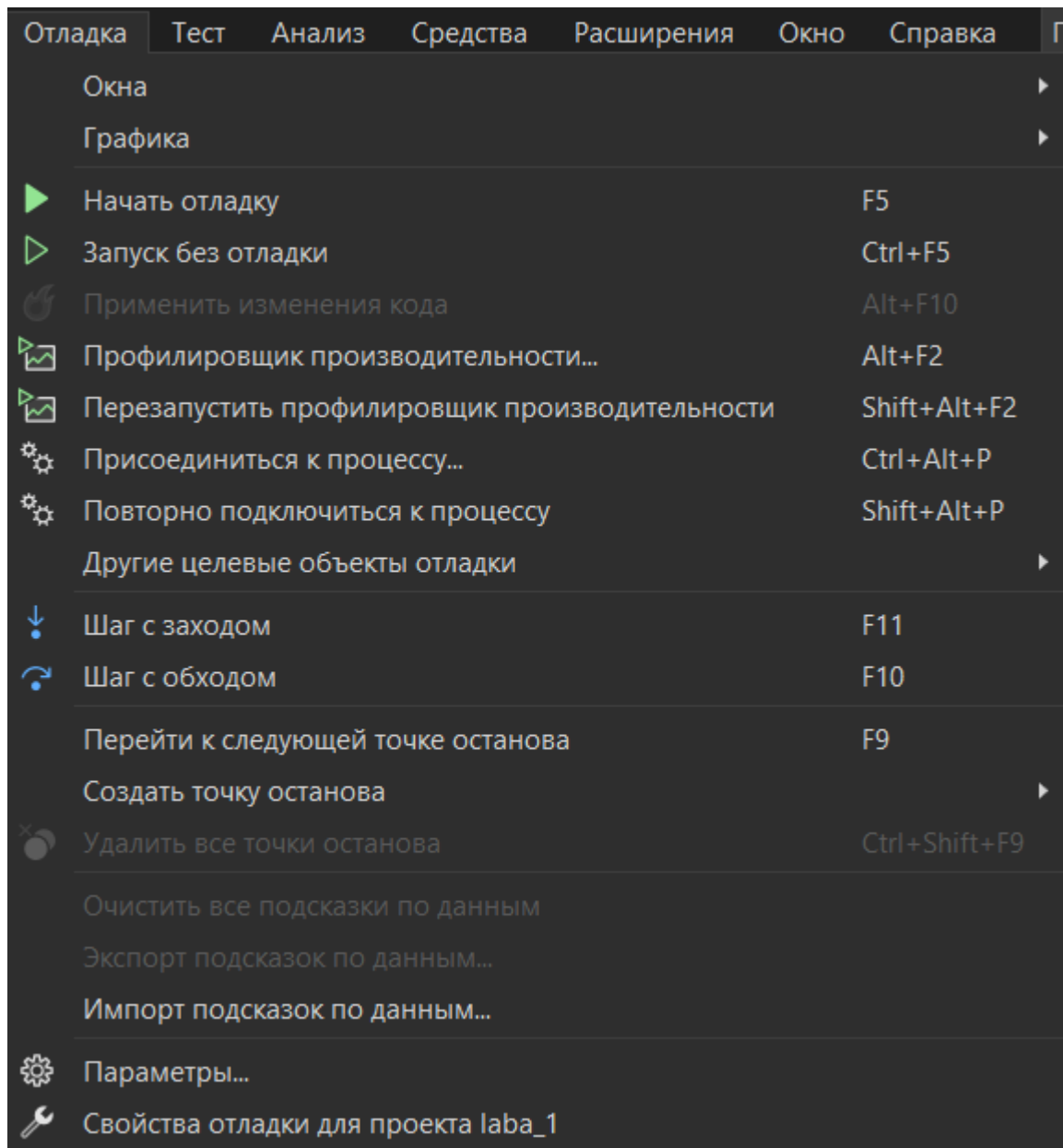


Рисунок 14 – Подменю отладки

Так будет выглядеть окно основной программы в режиме отладки:

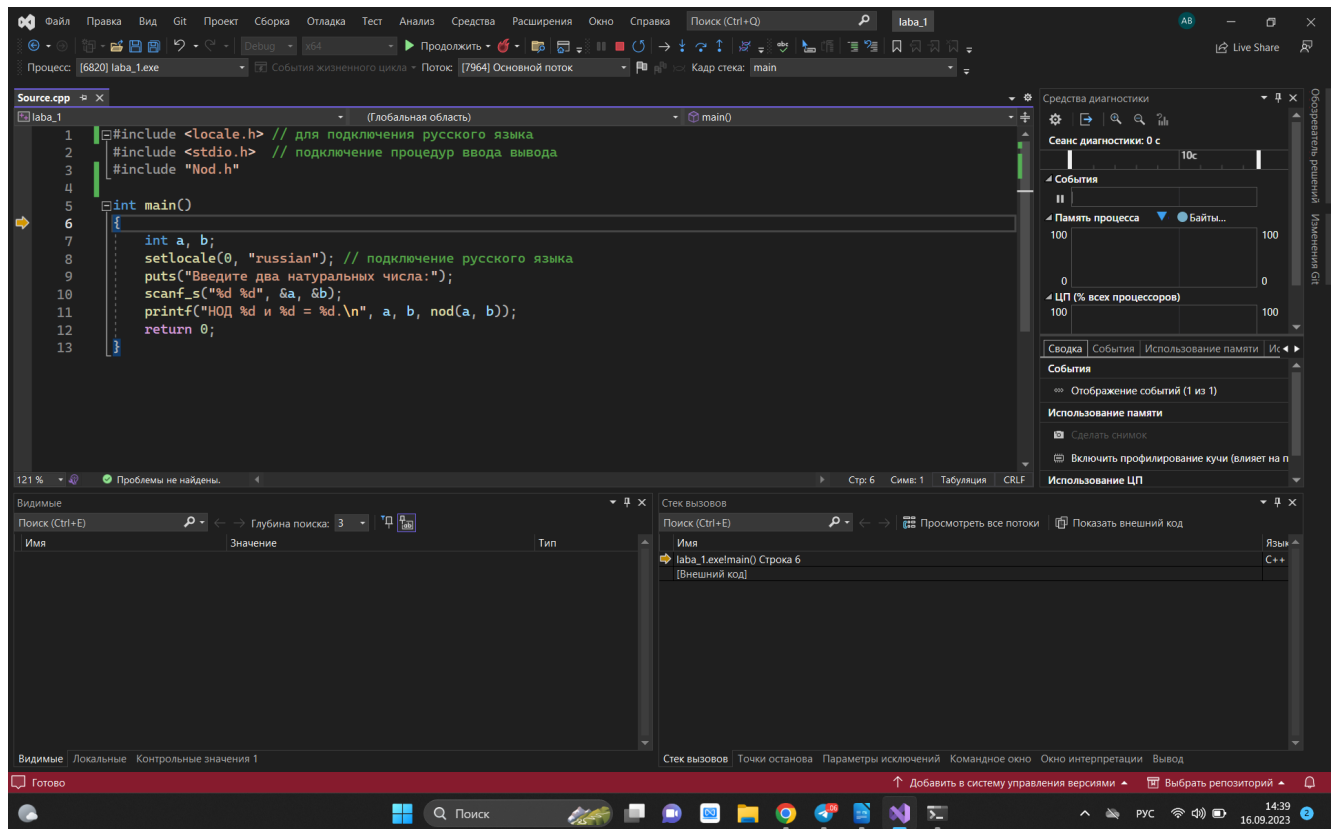


Рисунок 15 – Окно основной программы при запуске в режиме отладки

Так будет выглядеть подменю отладки в режиме отладки:

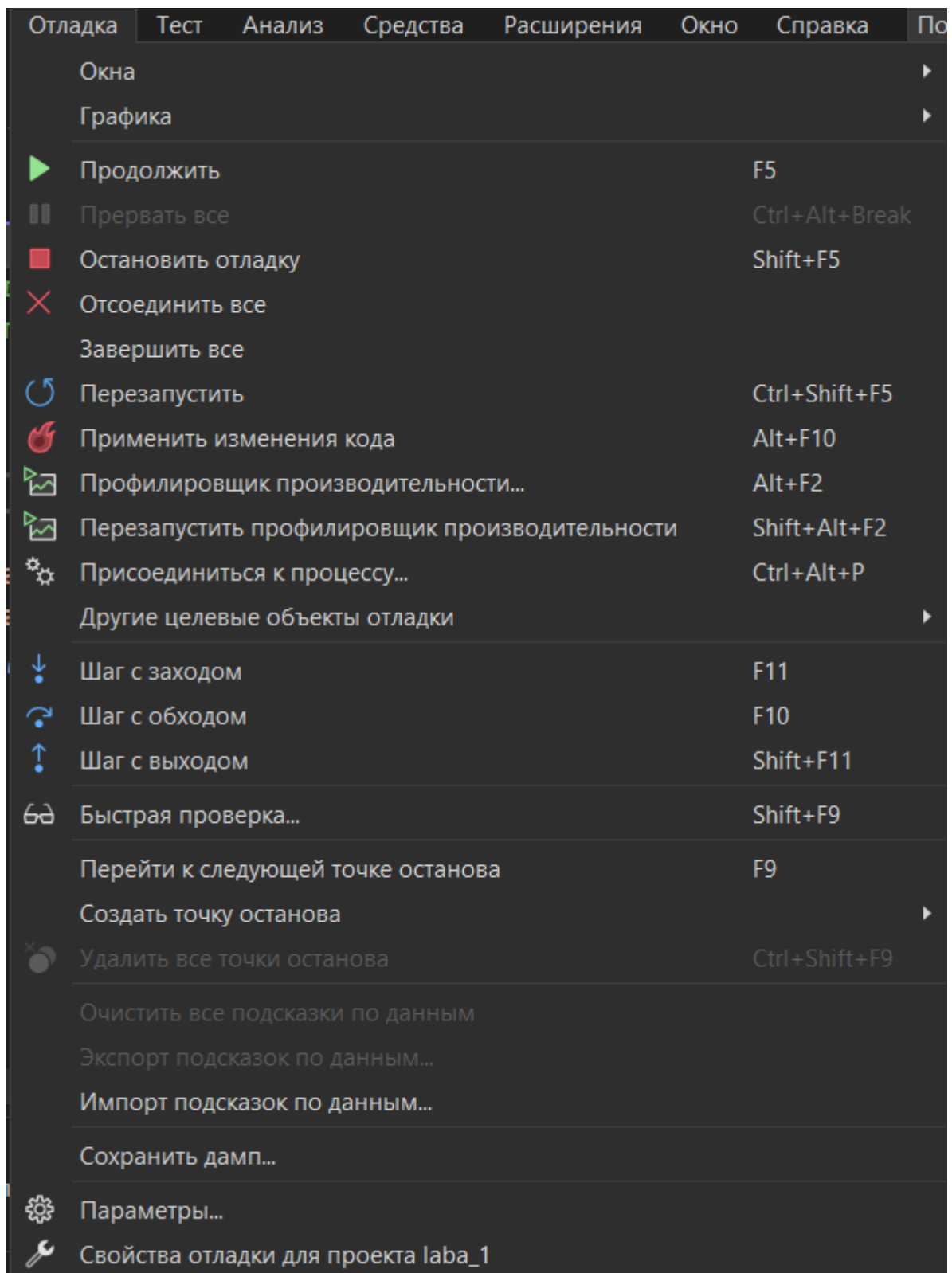


Рисунок 16 – Подпункты меню в режиме отладки

Чтоб следить за изменением переменных в Visual Studio снизу расположены 3 окна: Видимые, Локальные и Контрольные значения 1. Они используются для разных сценариев отладки.

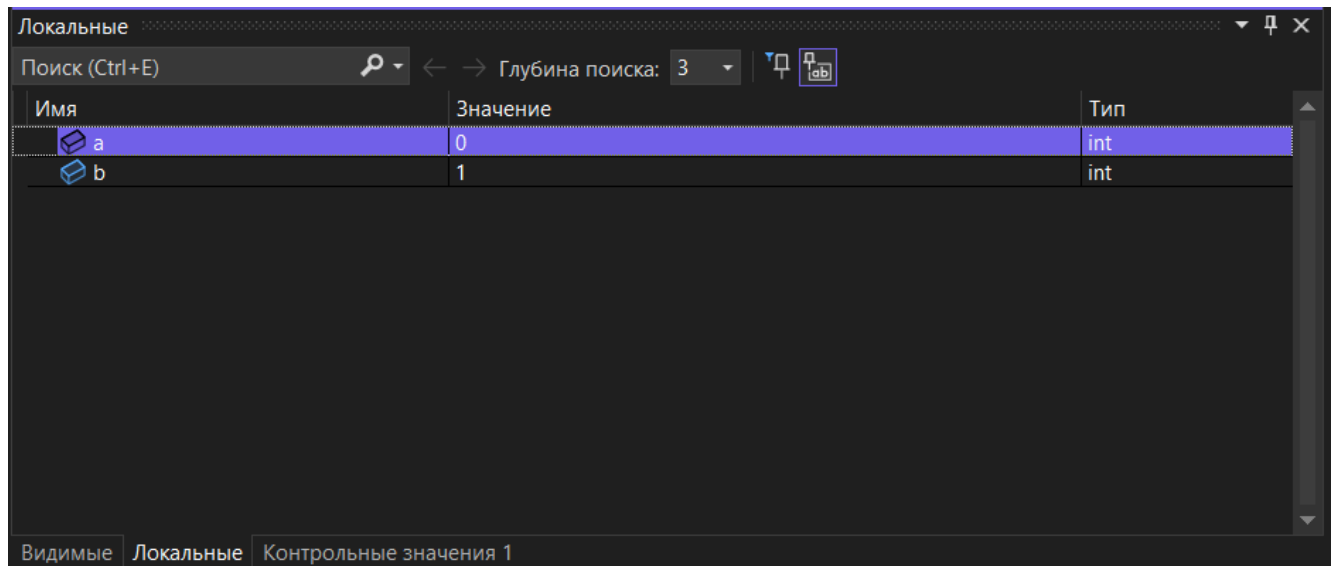


Рисунок 17 – Окно Локальные

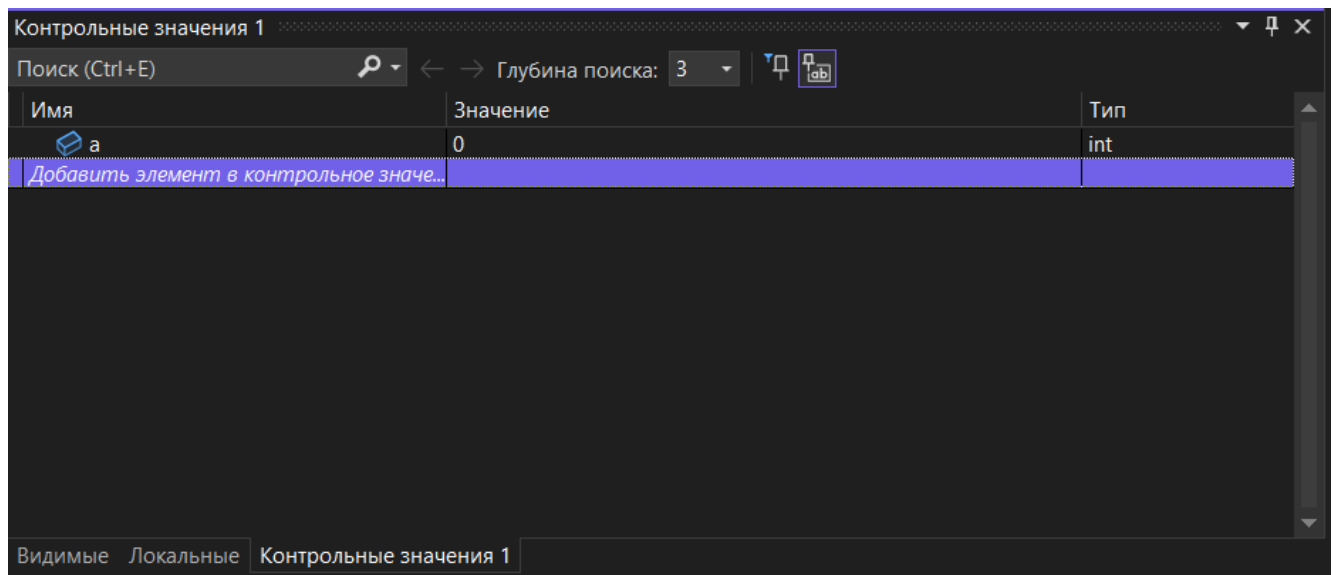


Рисунок 18 – Окно Контрольные значения 1

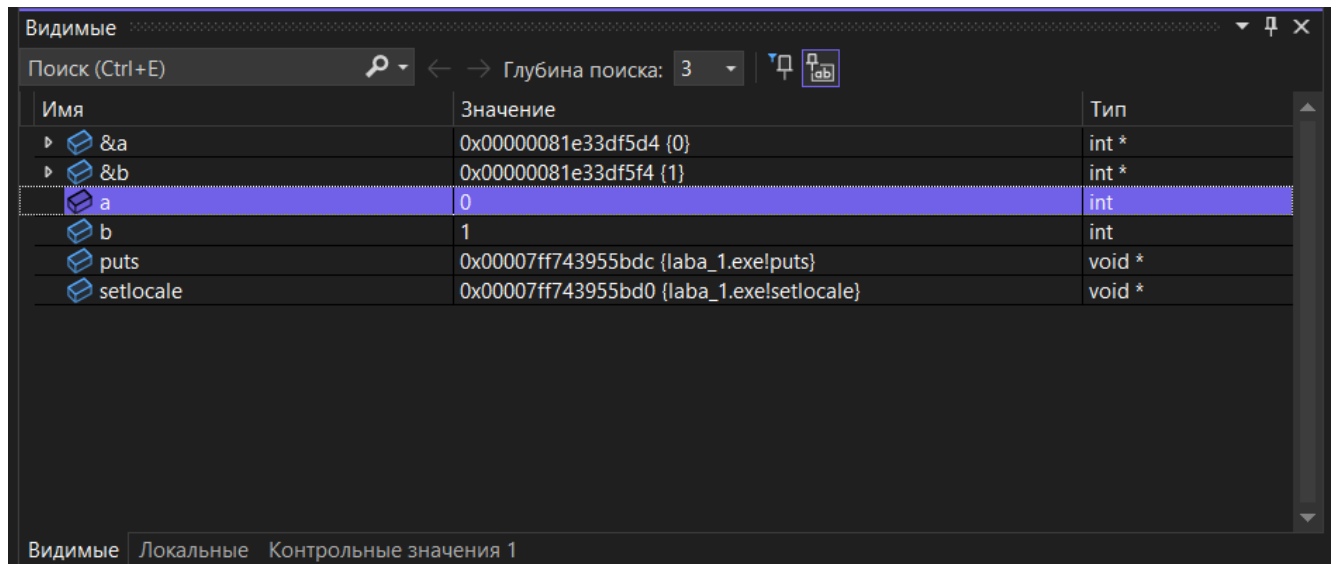


Рисунок 19 – Окно Видимые

Чтобы программа остановилась на нужном месте, необходимо поставить точку останова перед выполнением интересующего оператора:

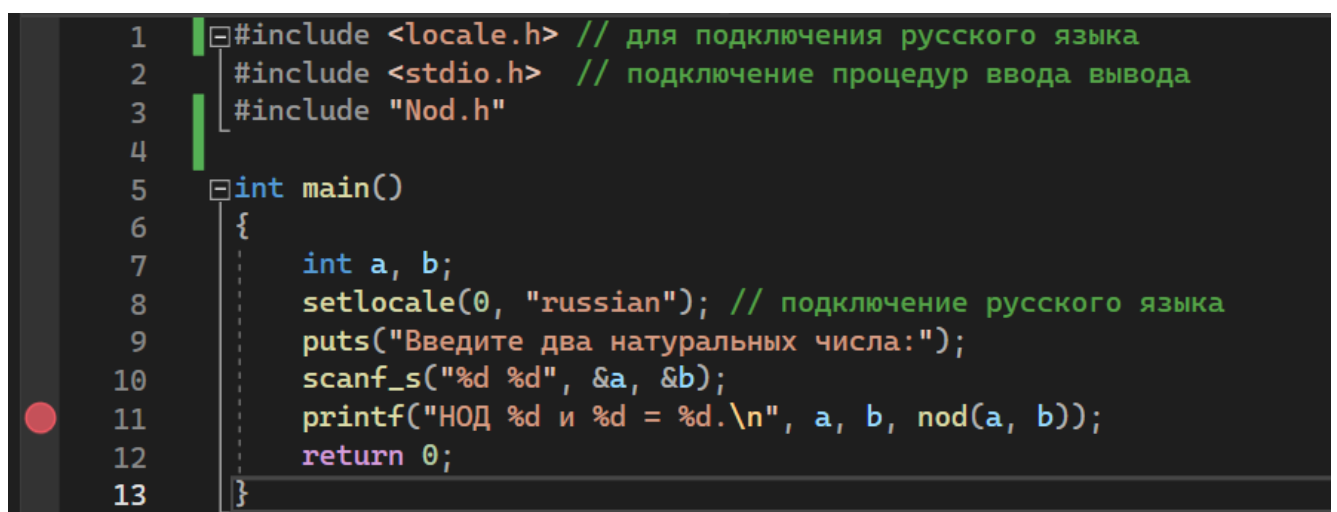


Рисунок 20 – Отметка точки останова в программе

Итак, для создания консольного приложения в среде Visual Studio с вводом и выводом на консоль необходимо:

1. Создать пустой проект.
2. Создать исходный файл в расширении .cpp.
3. Если есть предварительно откомпилированные заголовки, необходимо написать в коде программы `#include "pch.h"`. Такие заголовки можно отключить следующей последовательностью действий:

Проект->Свойства->Си/С++->Предварительно откомпилированные заголовки->Предварительно откомпилированный заголовок: Не использовать.

4. Для отмены проверки можно отключить опции контроля:
5. 1) **Проект->Свойства->Си/С++->Создание кода->Проверка безопасности: Отключить проверку безопасности (/GS-).**
2) **Проект->Свойства->Си/С++->Общие->Проверка SDL: Нет (/sdl-).**
6. Для того, чтобы консоль после выполнения программы не закрывалась, необходимо указать выполнить следующие действия:
Проект->Свойства->Компоновщик->Система->Подсистема: Консоль
7. Написать код программы.
8. Запустить в режиме отладки или без неё.
9. Ввести данные в всплывающем окне или увидеть результат выполнения программы.

Часть 2.

Наглядное изображение схемы алгоритма нахождения наибольшего общего делителя.

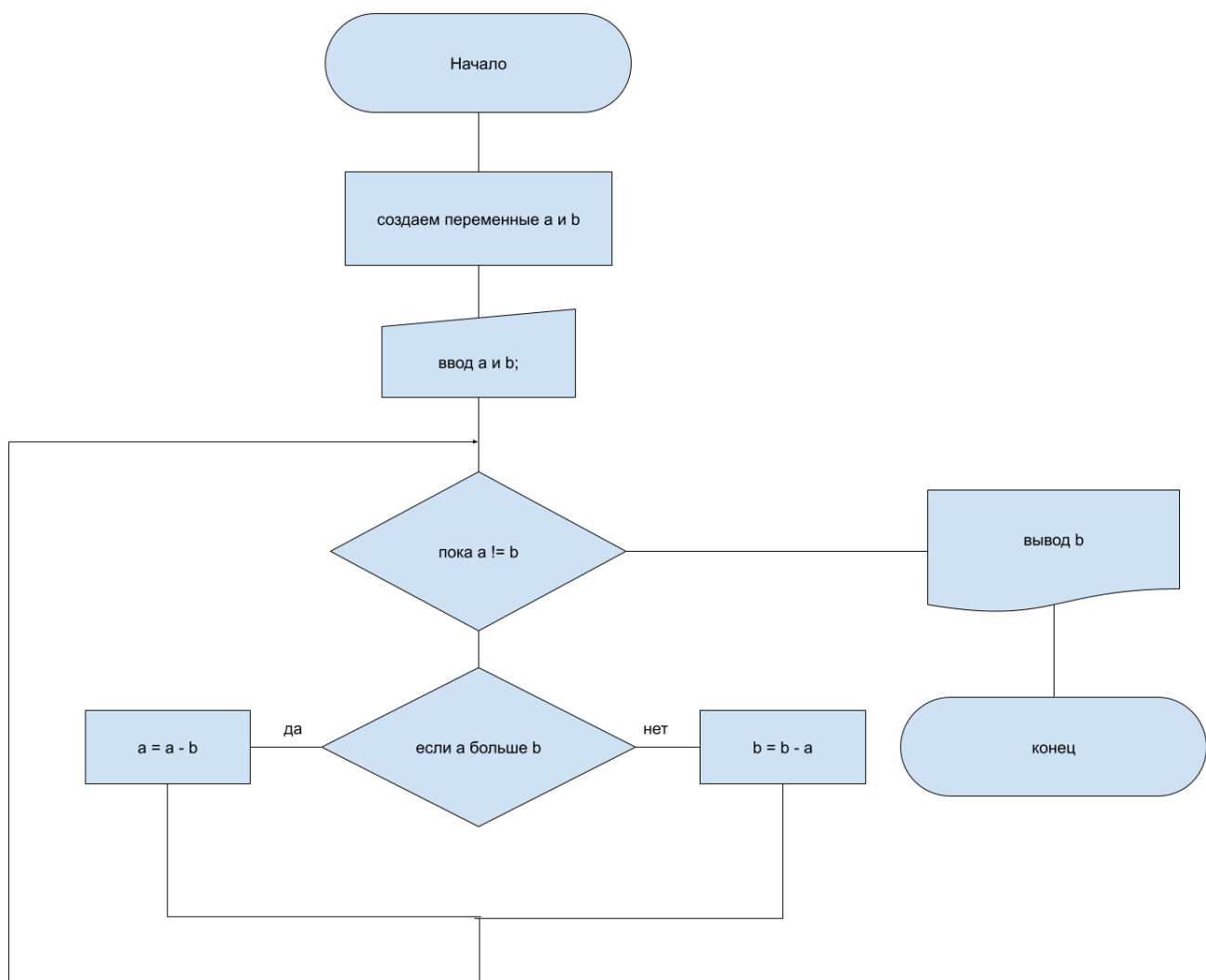


Рисунок 1 – схема алгоритма нахождения наибольшего общего делителя в Google Drawings.

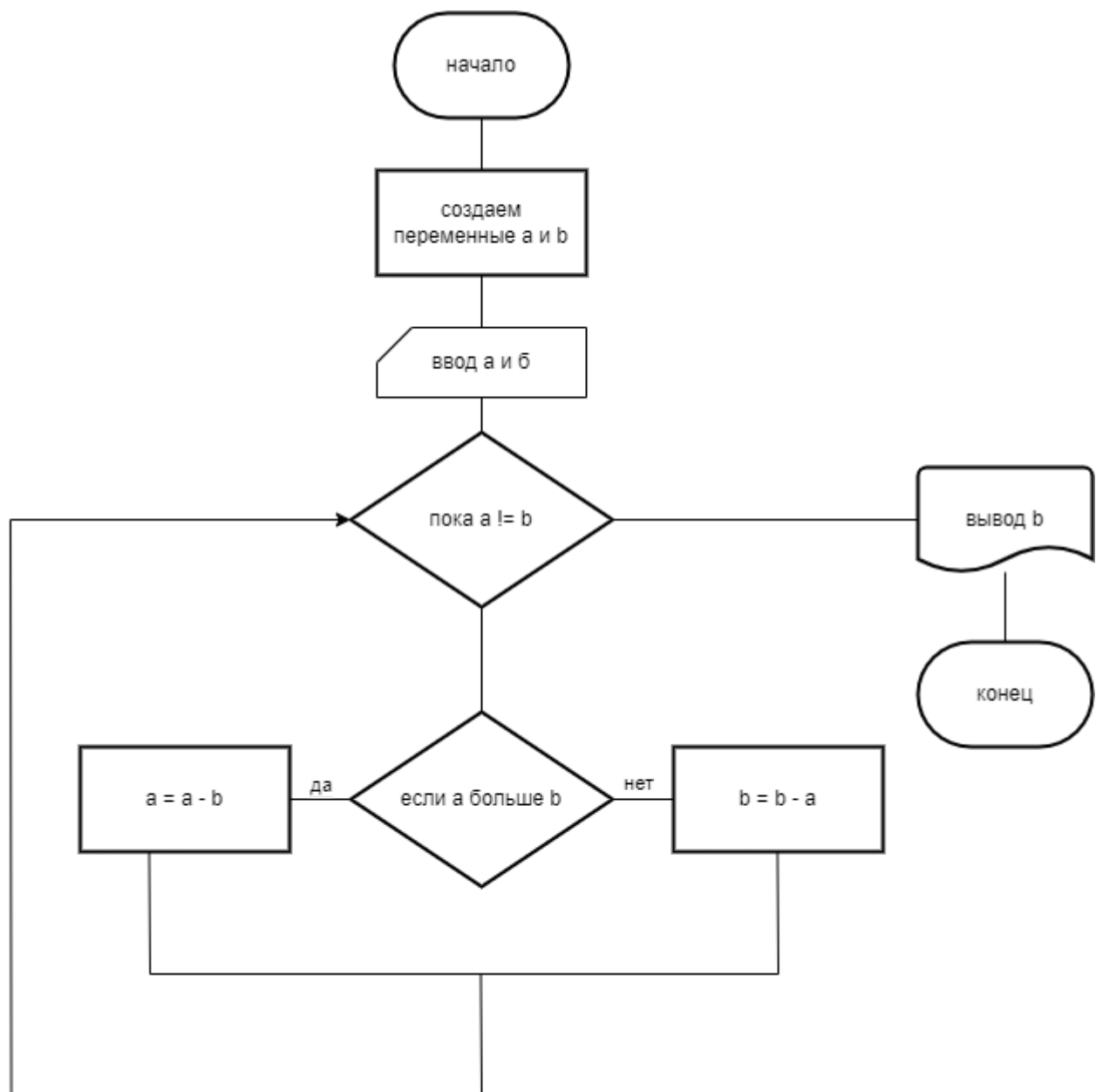


Рисунок 2 – Схема алгоритма нахождения наибольшего общего делителя в Draw.io

Вывод: В ходе лабораторной работы я научился создавать консольные приложения на ввод и вывод в среде разработки Visual Studio Community 2022, устанавливать параметры запуска программы, создавать исходные и заголовочные файлы и взаимодействовать с ними, создавать модули, подключать заголовочные файлы. Запускать программу в режиме отладки. Также научился пользоваться Google Drawings и Draw.io для создания схем алгоритма.