

Федеральное государственное автономное образовательное учреждение высшего
образования



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

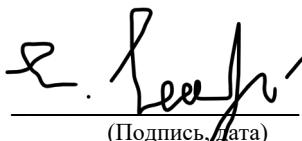
ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ
КАФЕДРА КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)


О т ч е т

по лабораторной работе № 1

**Название лабораторной работы: Консольные приложения на языке C++
в среде Qt Creator.**

Дисциплина: Алгоритмизация и программирование.

Студент гр. ИУ6-13Б  **С.М. Соболев**
(Подпись, дата) (И.О. Фамилия)

Преподаватель  15.09.25 **О.А. Веселовская**
(Подпись, дата) (И.О. Фамилия)

Часть 1. Консольные приложения на языке C++ в среде программирования Qt Creator.

Цель работы: научиться пользоваться средой Qt Creator.

Часть 1.1 Создание однофайлового проекта.

Цель: научиться создавать и работать с однофайловым проектом.

Была создана заготовка консольного приложения на языке программирования C. Проект был назван Rpnod и размещен в папке Документы.

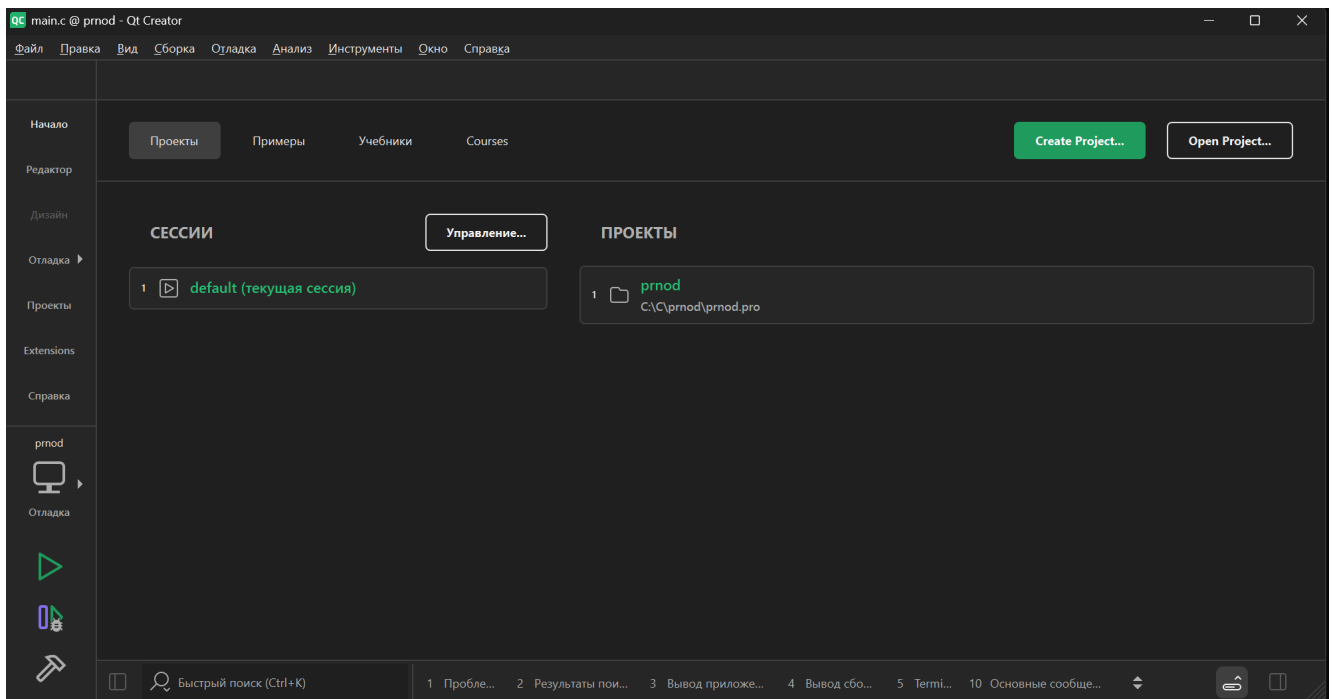


Рисунок 1 - Главное окно среды Qt Creator

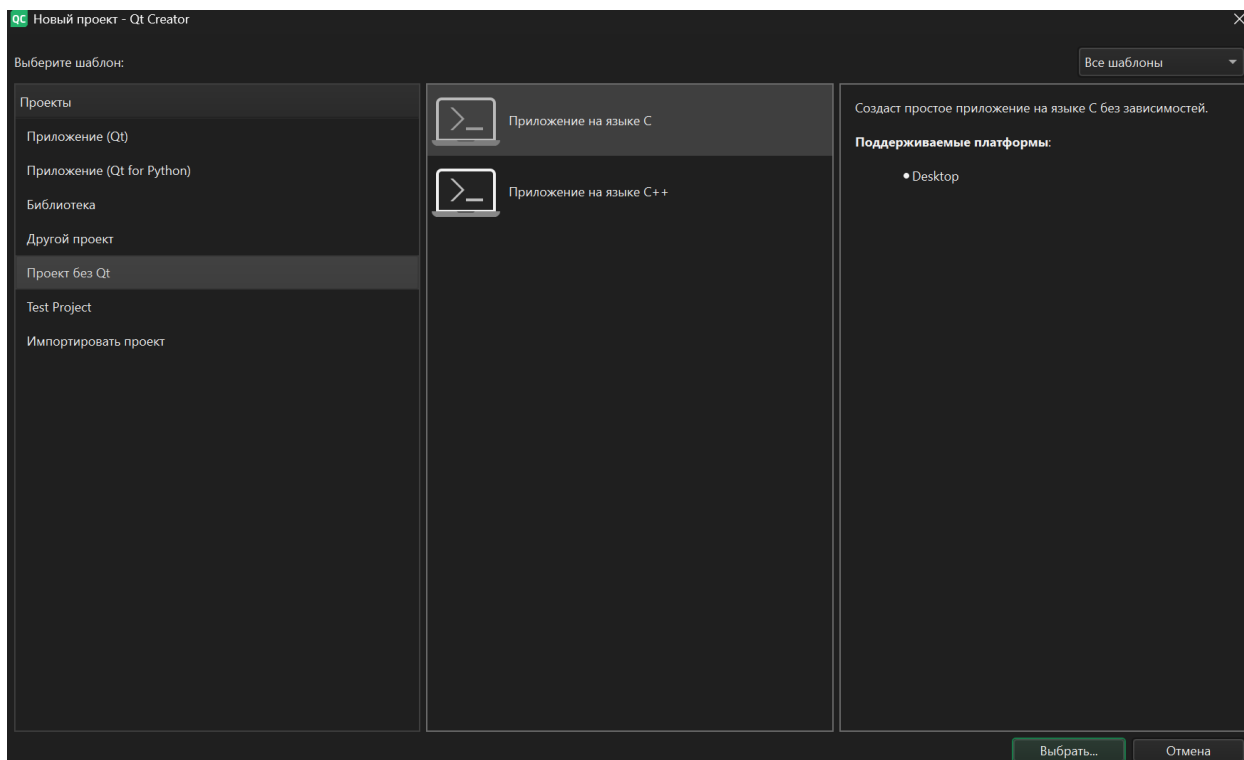


Рисунок 2 - Вид окна выбора типа и шаблона проекта.

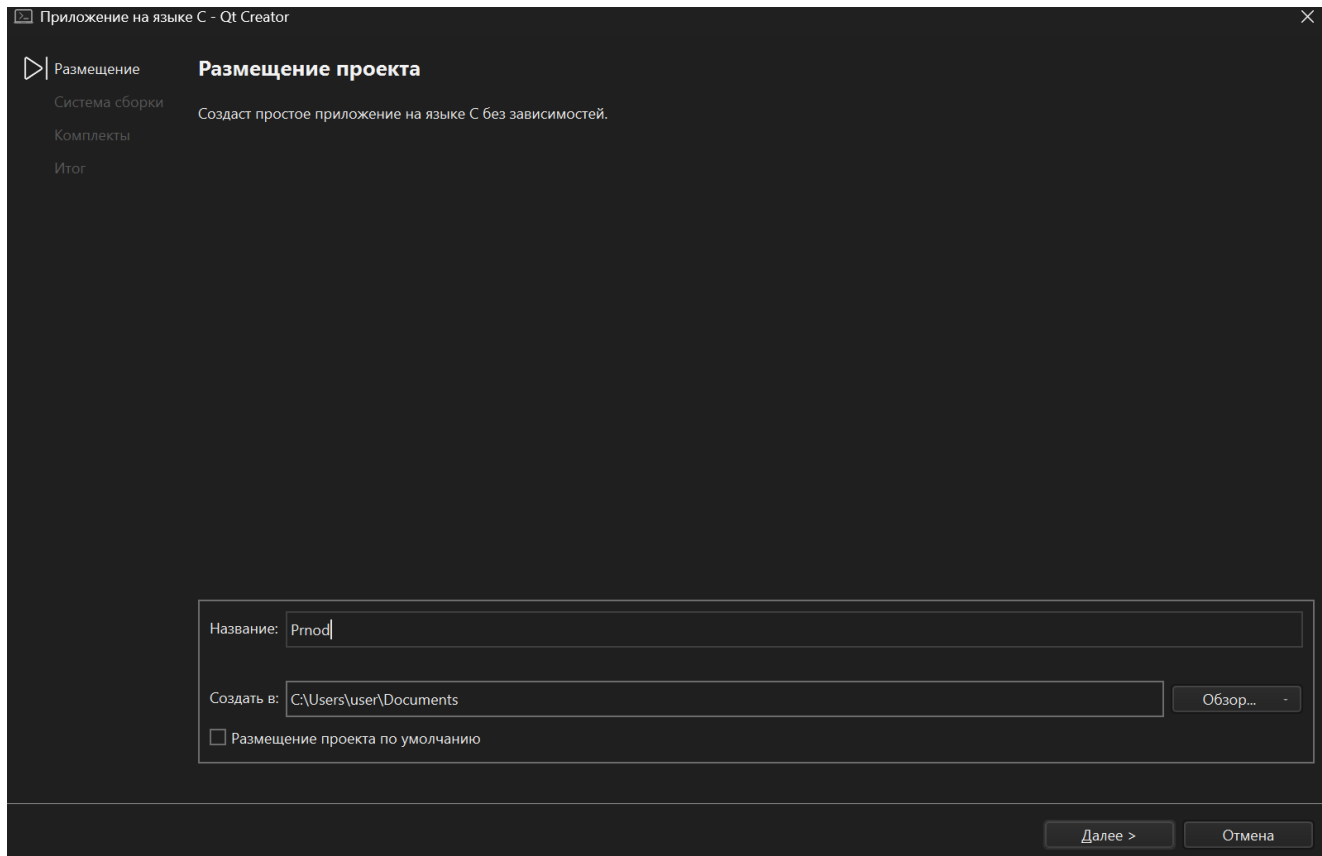


Рисунок 3 - Вид окна выбора типа и шаблона проекта.

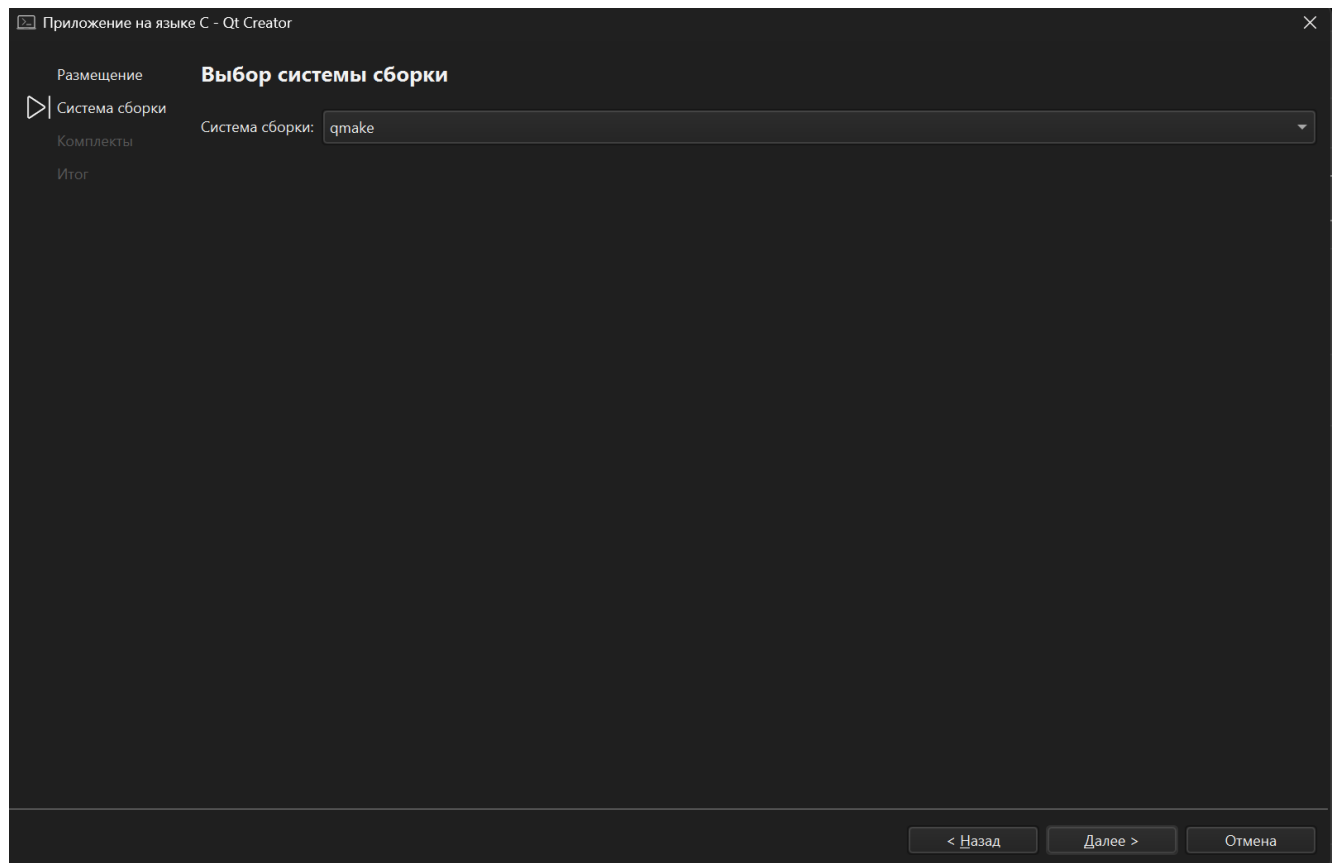


Рисунок 4 - Вид окна выбора системы сборки.

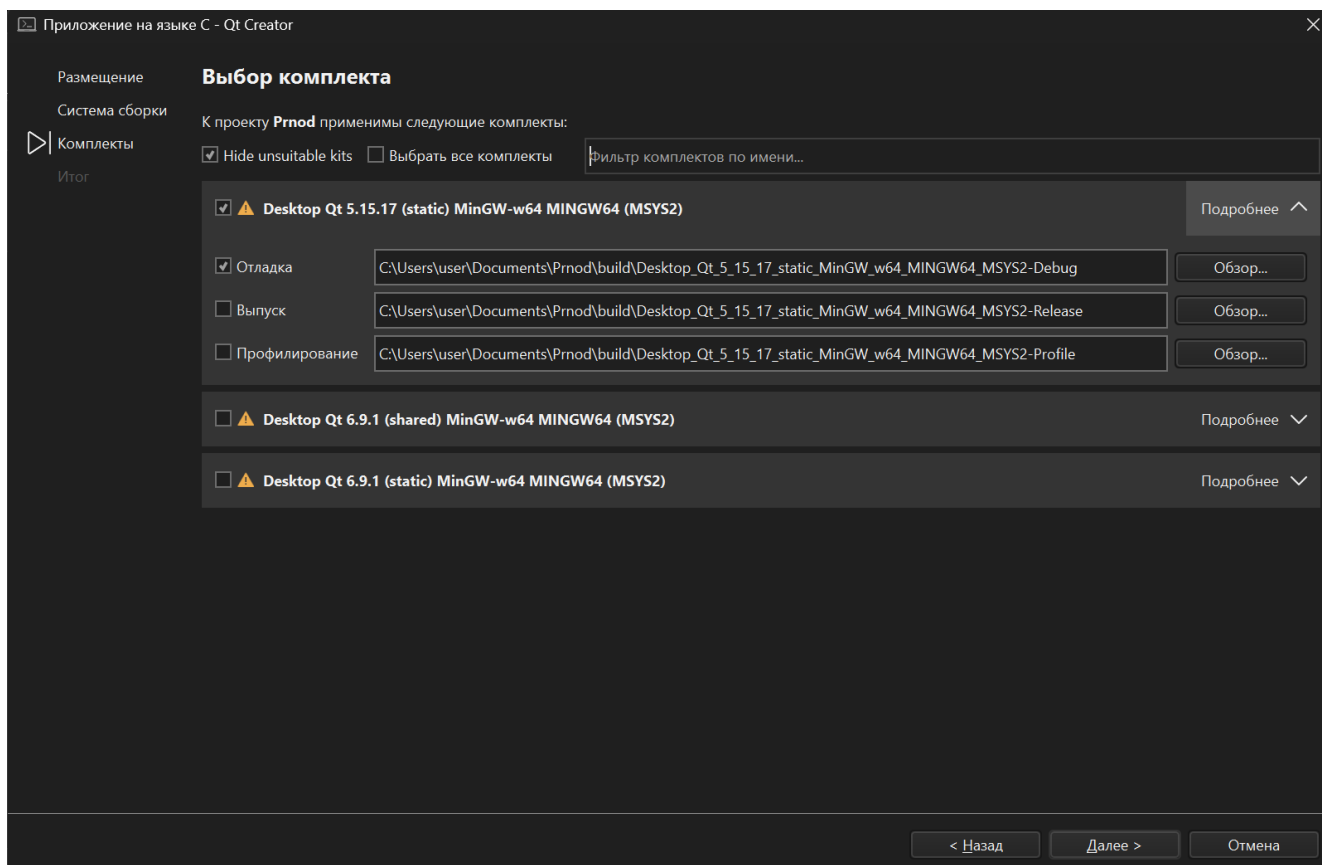


Рисунок 5 - Вид окна выбора комплектов сборки.

После этого был настроен запуск программы в терминале (рисунок 6).

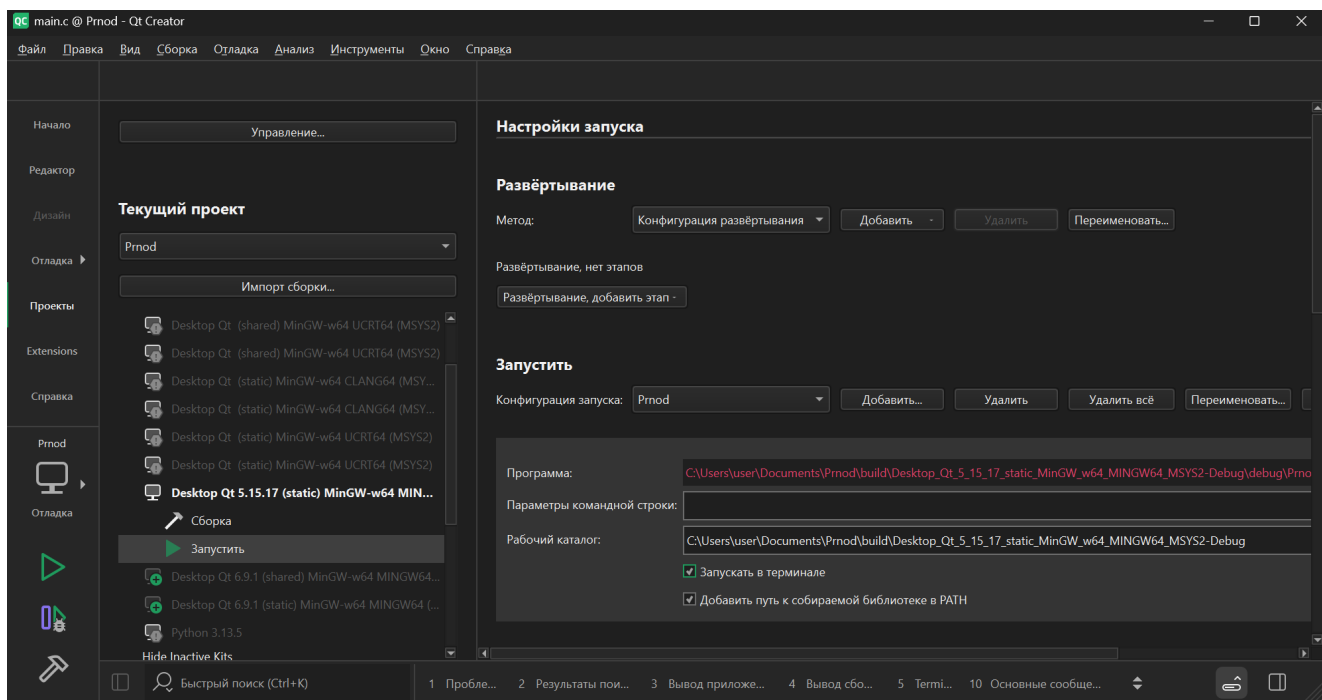
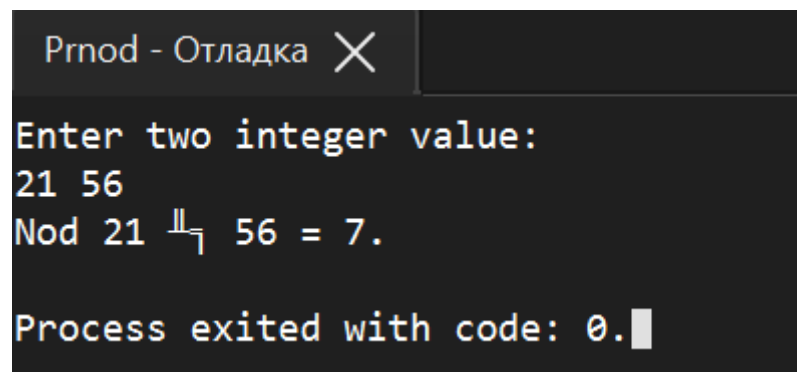


Рисунок 6 – Настройка запуска проекта.

Затем в окно редактора была введена программа вычисления наибольшего общего делителя двух целых чисел (Рисунок 7). Был произведен запуск программы (Рисунок 8).

```
1  #include <stdio.h> // подключение процедур ввода вывода
2  int nod(int x, int y) {
3      while (x!=y)
4          if (x>y) x=x-y;
5          else y=y-x;
6      return y;
7  }
8  int main(){
9      int a,b;
10     puts("Enter two integer value:");
11     scanf("%d %d",&a,&b);
12     printf("Nod %d и %d = %d.\n",a,b,nod(a,b));
13     return 0;
14 }
15
```

Рисунок 7 – Код программы для вычисления НОД.



```
Prnod - Отладка X
Enter two integer value:
21 56
Nod 21 и 56 = 7.
Process exited with code: 0.
```

Рисунок 8 – Запуск программы.

Вывод: узнал особенности создания консольных приложений а среде Qt Creator, получил навыки работы с приложением и его настройкой для корректной работы.

Часть 1.2 Многофайловый проект.

Цель: научиться создавать и работать с многофайловыми проектами.

В папке проекта был создан заголовочный файл Nod.h (Рисунок 9-10). В нем же было добавлено определение функции pod (Рисунок 11).

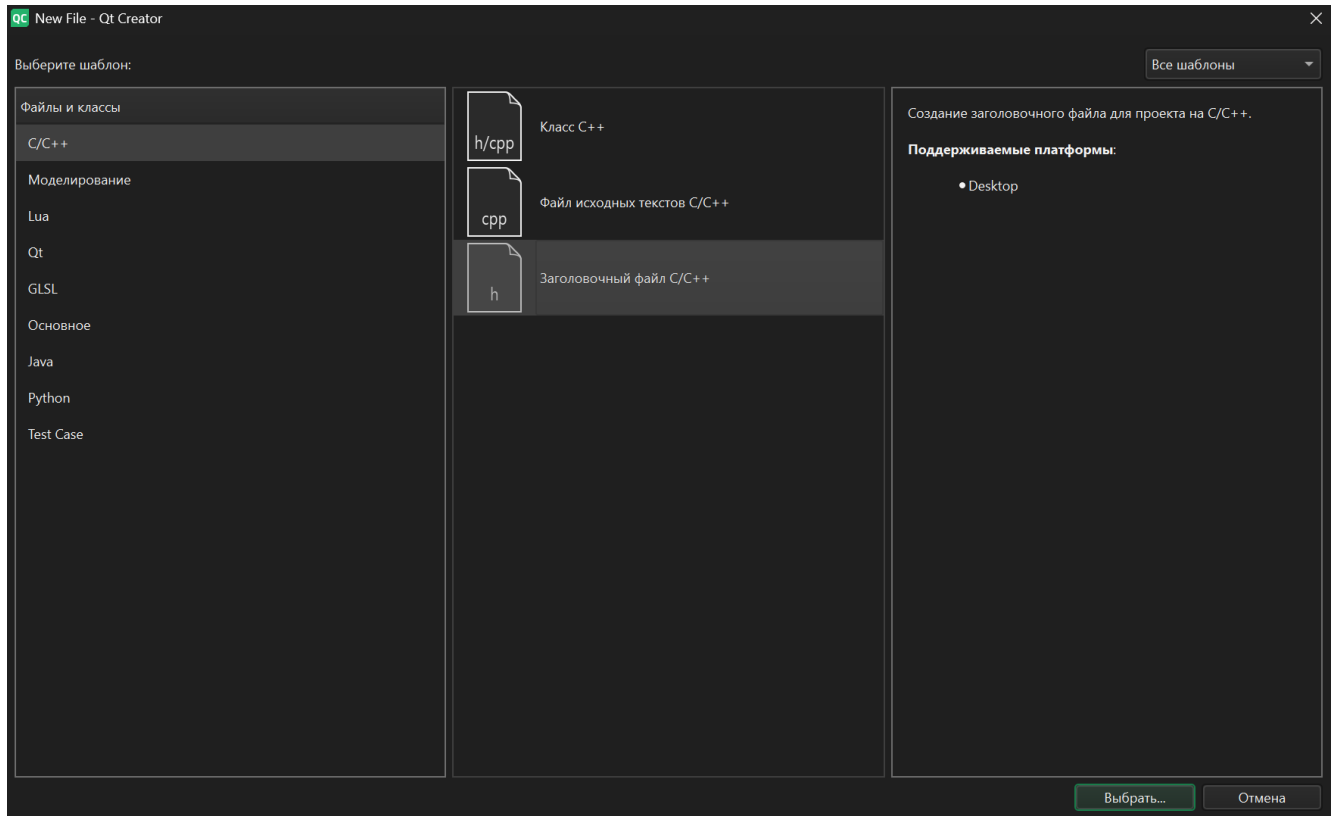


Рисунок 9 – Создание заголовочного файла.

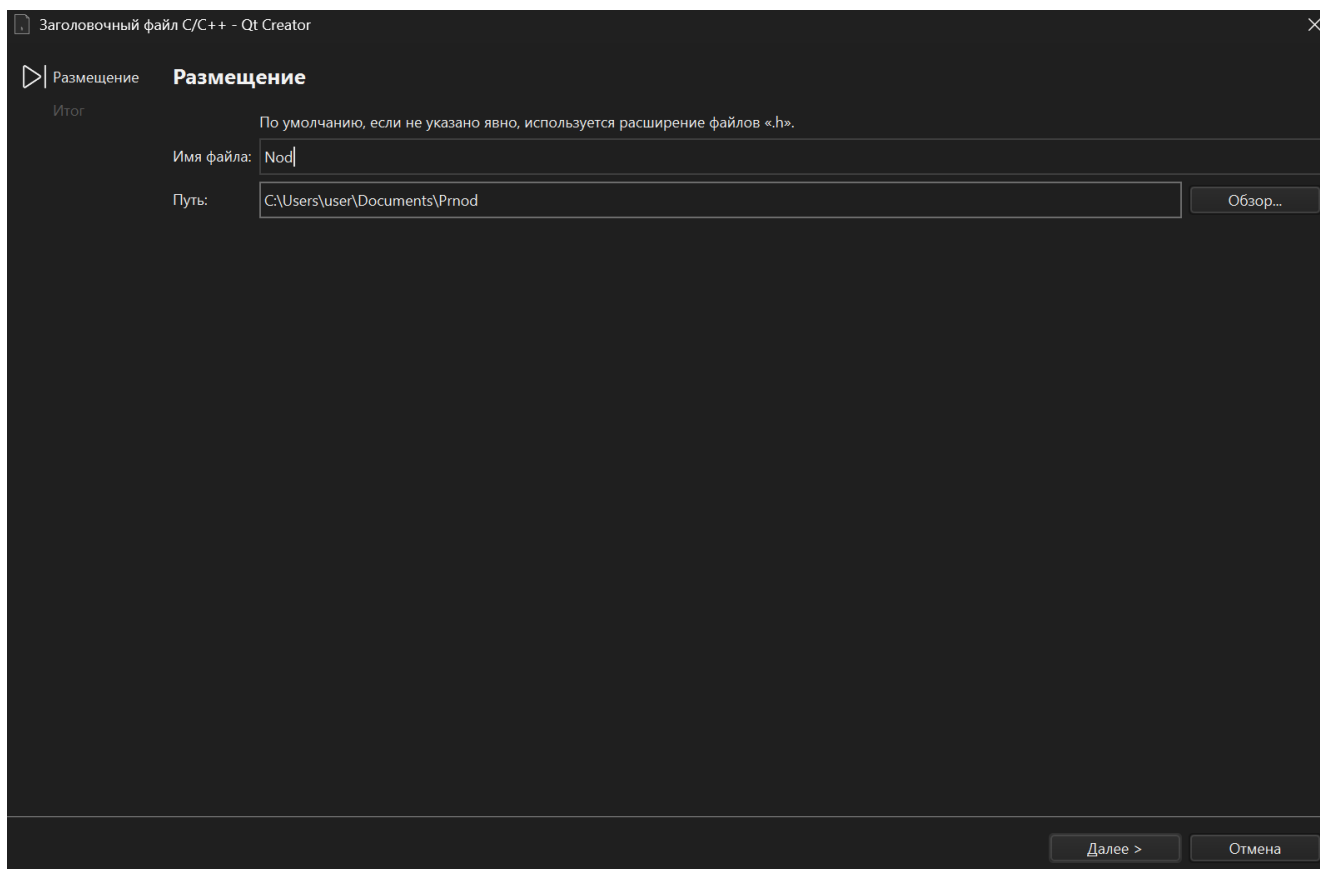


Рисунок 10 – Ввод имени заголовочного файла.

```

1  #ifndef NOD_H
2  #define NOD_H
3  int nod(int x, int y)
4  #endif // NOD_H
5

```

Рисунок 11 – Определение функции.

Таким же образом был создан второй файл Nod.c для реализации модуля. В файл перенесен текст функции `nod` и добавлена строка подключения заголовочного файла (Рисунок 12).

```

1  #include "Nod.h"
2  int nod(int x, int y){
3      while (x!=y)
4          if (x>y) x=x-y;
5          else y=y-x;
6          return y;
7      }
8

```

Рисунок 12 – Код функции `nod` в файле.

Вместо функции `nod` в файле `main.c` была добавлена ссылка на заголовочный файл `Nod.h`. Вид текста основной программы *main* после корректировки приведен на рисунке 13.

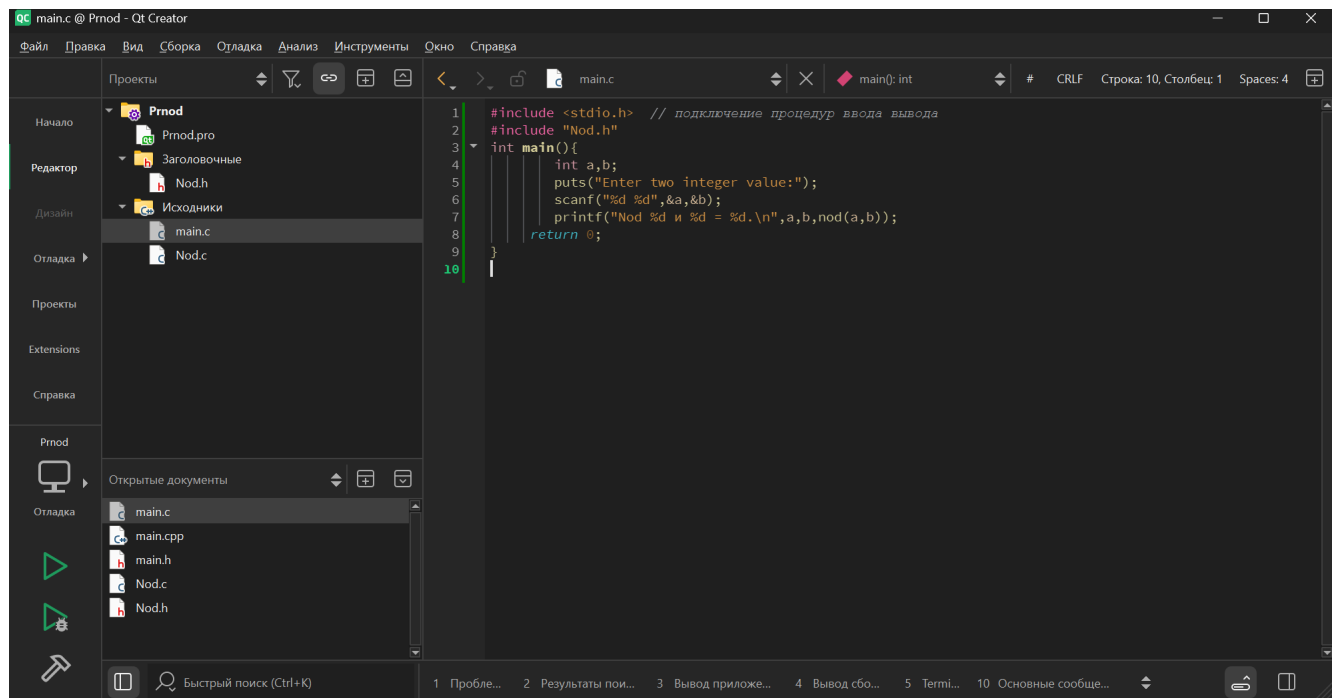


Рисунок 13 – Вид окна после приложения корректировки *main*.

Программа после корректировки была проверена с теми же данными, на экране показывается тот же самый результат (Рисунок 14).

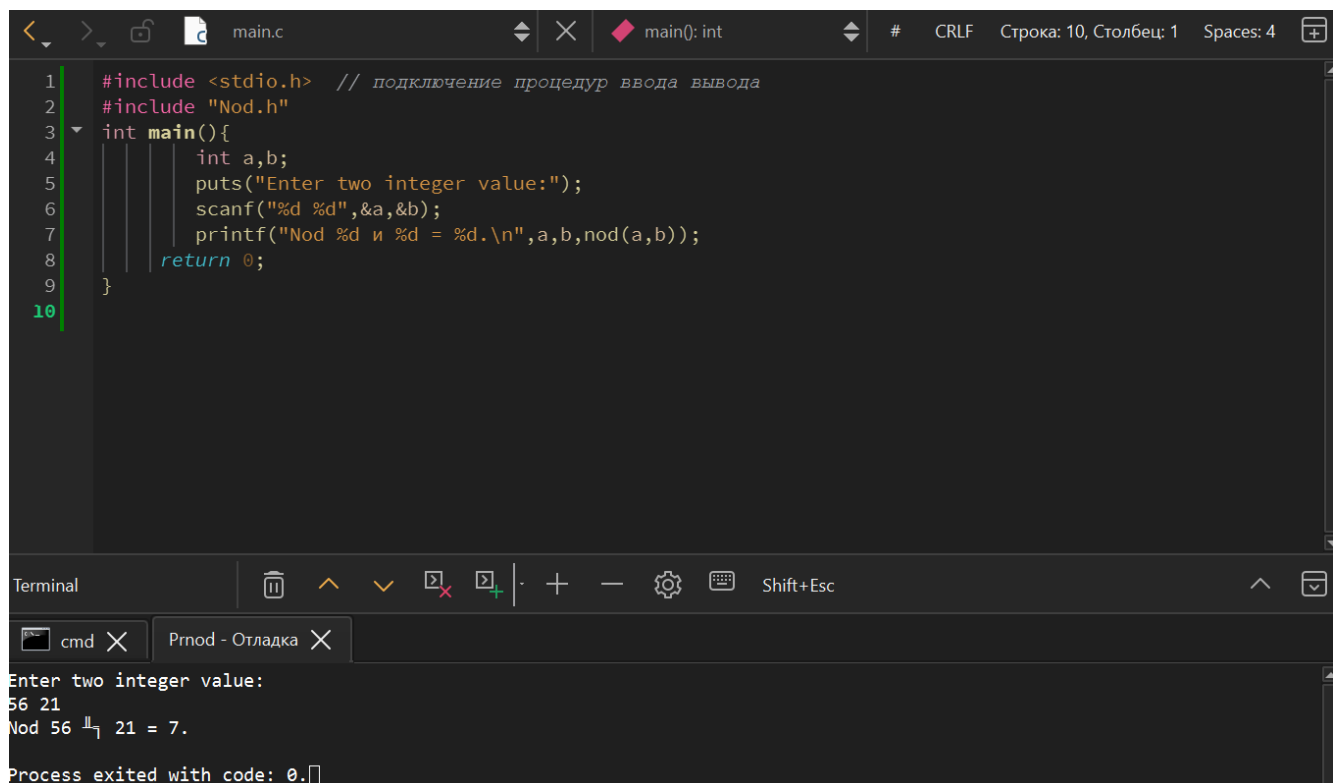


Рисунок 14 - Вид окна приложения после запуска многофайлового проекта.

Вывод: были получены навыки работы с многофайловым проектом.

Часть 1.3. Отладка.

Цель: научиться пошагово отлаживать программу.

Была начата отладка программы при помощи пункта в верхнем меню **Отладка** и подпункта **Запустить и встать на main**.

С помощью клавиши F10 был выполнен шаг программы с вводом.

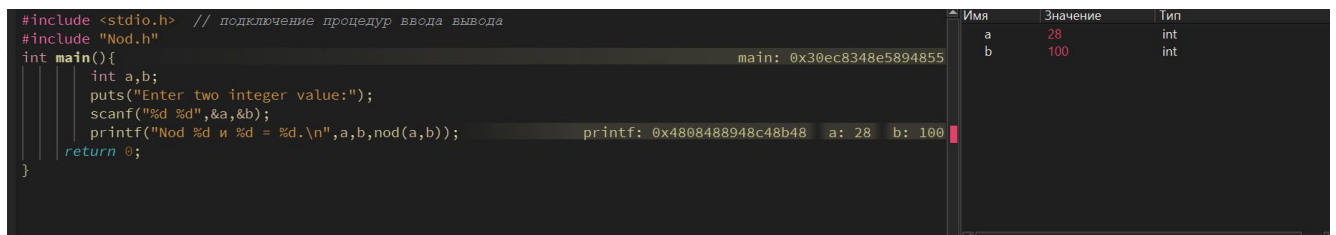


Рисунок 15 – Отладка программы.

Была добавлена переменная во второе окошко, кликнув правой кнопкой мыши по нему и выбрав пункт **Добавить вычисляемое значение**.

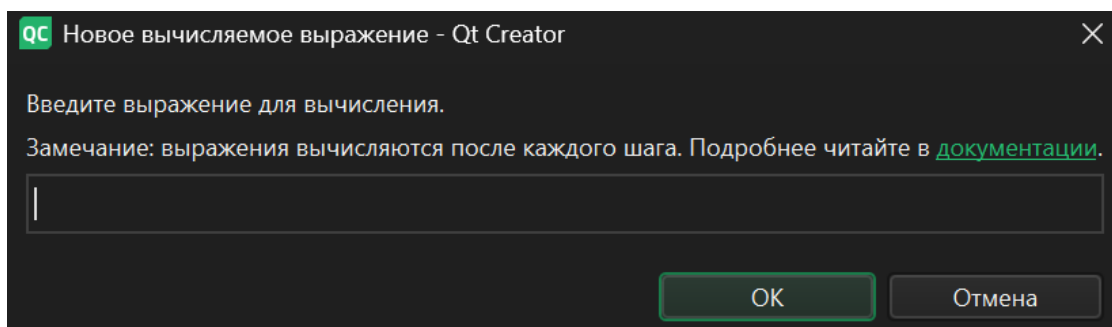


Рисунок 16 – Окно для вычисления переменной/выражения.

При нажатии на левое поле, была добавлена красная точка для остановки отладчика в данной строке.

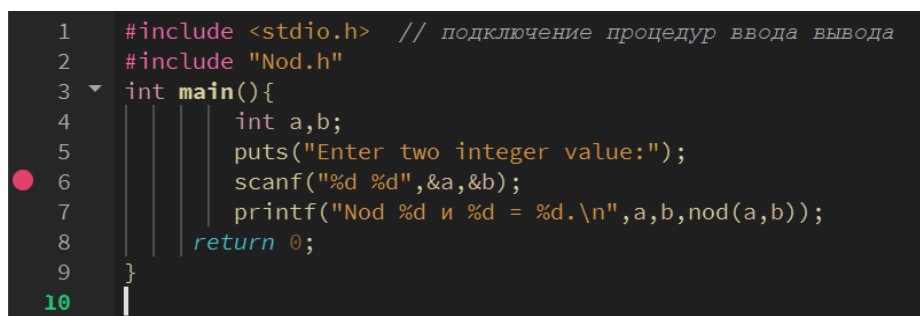


Рисунок 17 – Создание точки останова отладчика.

Вывод: были получены навыки с отладчиком и пошаговой отладкой, с заходом в подпрограмму, точек останова и добавление просматриваемых переменных.

Заключение: научился пользоваться средой Qt Creator с однофайловыми и многофайловыми проектами, получил навыки работы с отладчиком.

Часть 2. Создание схем в среде Microsoft Visio 2016 и Draw.io.

Цель работы: получить навыки создания схем и работы со средой Microsoft Visio 2016 и Draw.io.

Часть 2.1 Создание схемы в Microsoft Visio 2016.

Цель: создать схему алгоритма и овладеть базовыми навыками работы в Microsoft Visio 2016.

Первым шагом был запуск программы «Microsoft Visio 2016». В стартовом окне для создания нового документа был выбран шаблон «Простая схема» (Рисунок 18). Это действие привело к созданию пустого файла.

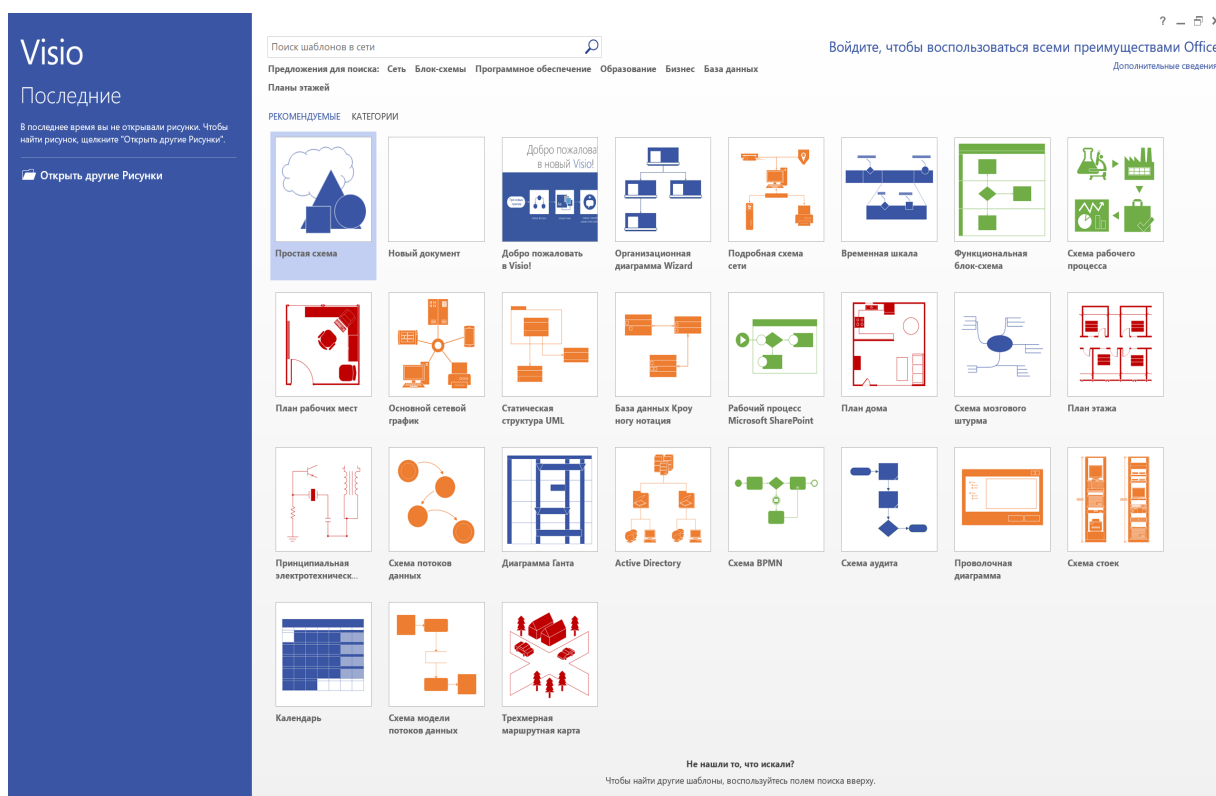


Рисунок 18 – Создание пустого документа в Microsoft Visio 2016

Основное наполнение схемы алгоритма было выполнено с помощью элементов из раздела «Фигуры». Базовые блоки находятся в категории «Простые фигуры» (Рисунок 19). Процесс добавления фигуры на страницу заключается в её перетаскивании из области элементов на рабочее поле. Для любой добавленной фигуры доступно изменение таких свойств, как размер, цвет фона и цвет линии, а также редактирование формы (например, степень закругления углов, толщина и ширина контура). Чтобы добавить текстовое описание, используется двойной щелчок левой кнопкой мыши по объекту. Форматирование текста осуществляется с помощью инструментов вкладки «Главная».

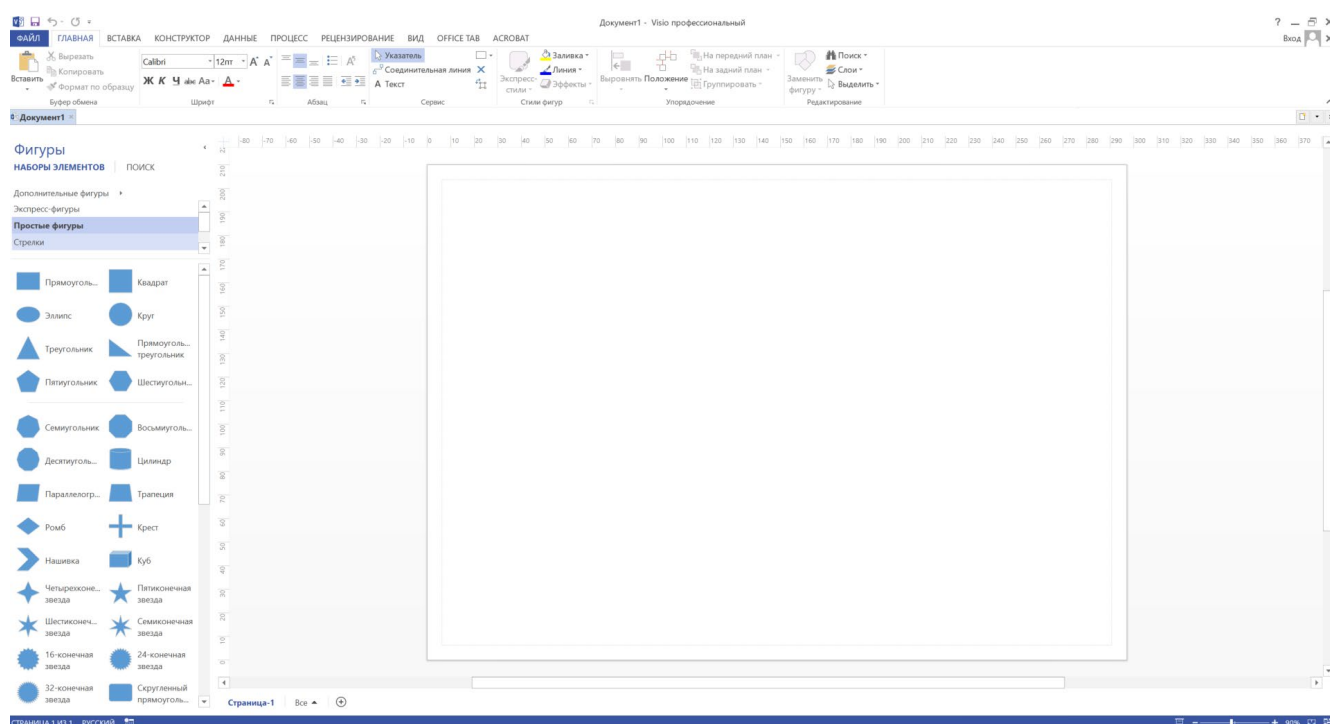


Рисунок 19 - Подраздел «Простые фигуры» Microsoft Visio 2016

На следующем этапе отдельные фигуры были соединены между собой линиями и стрелками. Для этого применялись элементы из подраздела «Стрелки». Чтобы нарисовать прямую линию, нужно выбрать инструмент «Прямая линия» и разместить его на холсте (Рисунок 20).

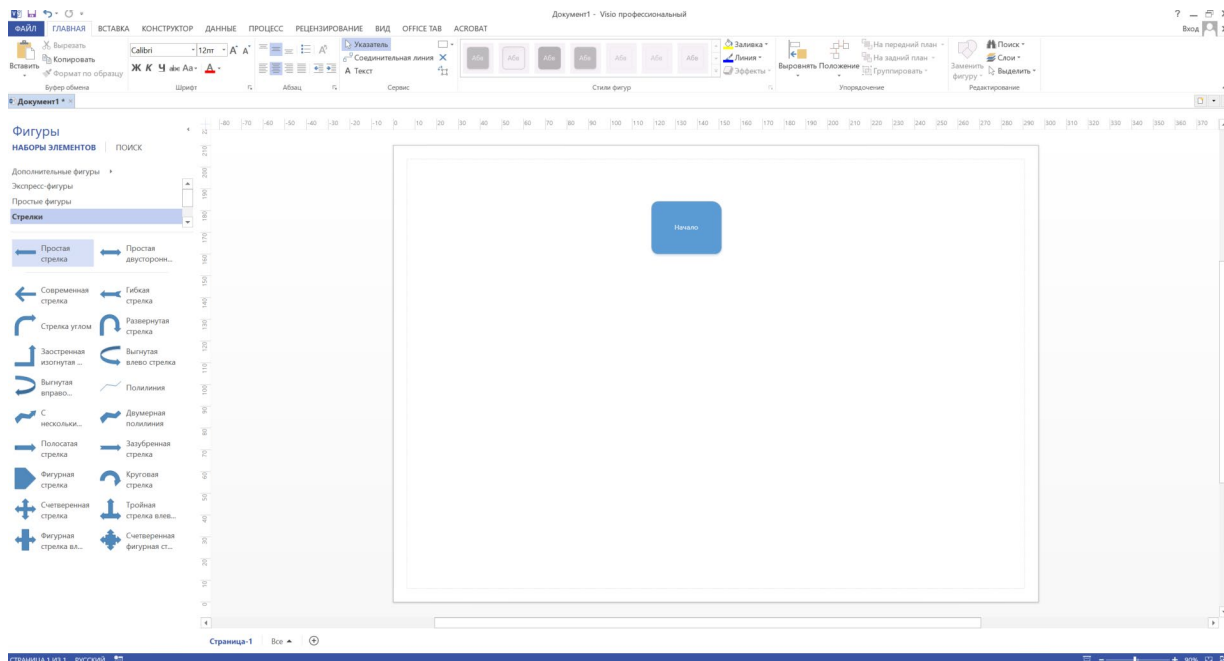


Рисунок 20 -Добавление линий Microsoft Visio 2016

Для обозначения направления потока в соответствующих местах добавлялись стрелки с помощью элемента «Линия со стрелкой». После размещения стрелки на схеме для нее можно задать ориентацию. Это делается через контекстное меню, вызываемое правой кнопкой мыши, где выбирается нужное направление: в обе стороны, влево или вправо . После завершения работы над схемой документ был сохранён через пункт меню «Файл» -> «Сохранить как...». В результате была получена готовая схема алгоритма, которая представлена на Рисунке 21.

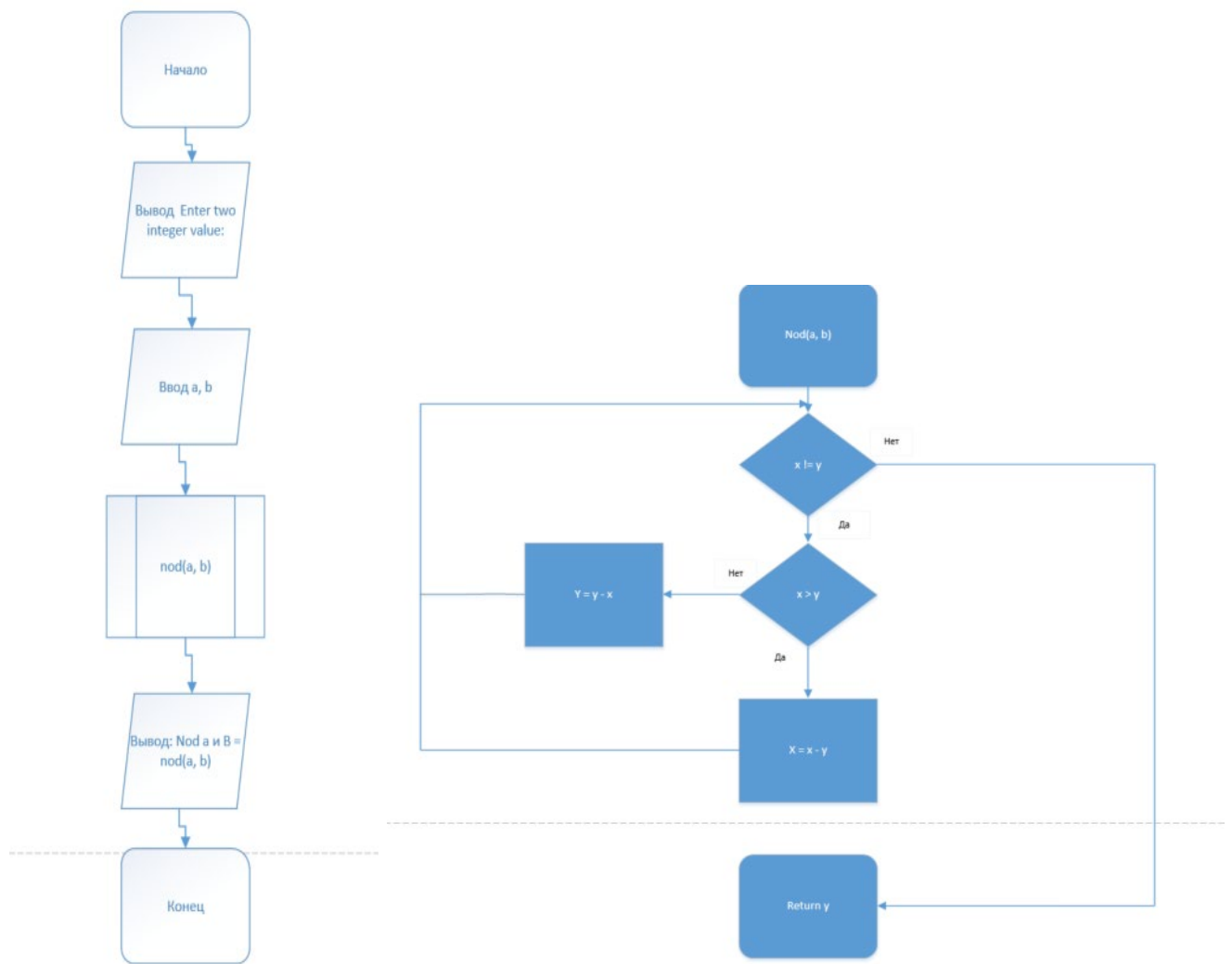


Рисунок 21 – Схемы созданные в Microsoft Visio 2016.

Вывод: была успешно разработана схема алгоритма и приобретены основные навыки использования инструментария «Microsoft Visio 2016».

Часть 2.2 Создание схемы в среде Draw.io.

Была запущена среда Draw.io (Рисунок 22). При помощи фигур в разделе «Общие» из левой панели были добавлены основные фигуры схемы. Двойным щелчком левой кнопкой мыши по фигуре можно изменить текст, находящийся внутри фигуры.

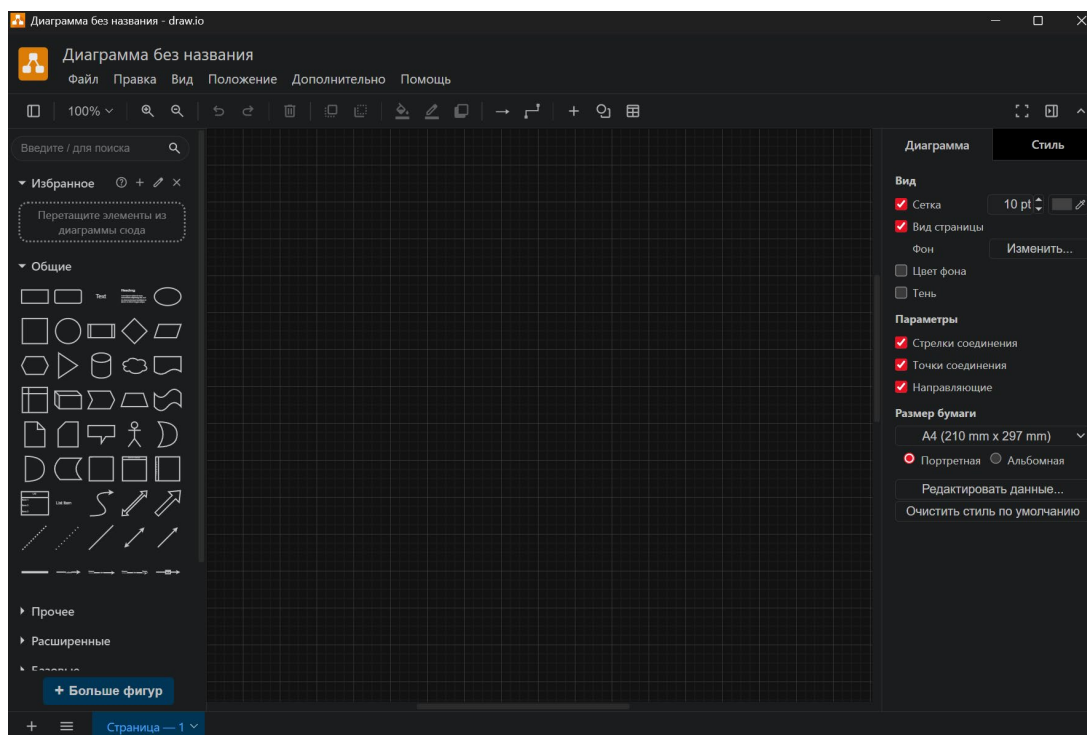


Рисунок 22 – Главное окно приложения.

С помощью нажатия левой кнопкой мыши на фигуры и выбора «Стрелки». Были проведены соединительные линии (Рисунок 23).

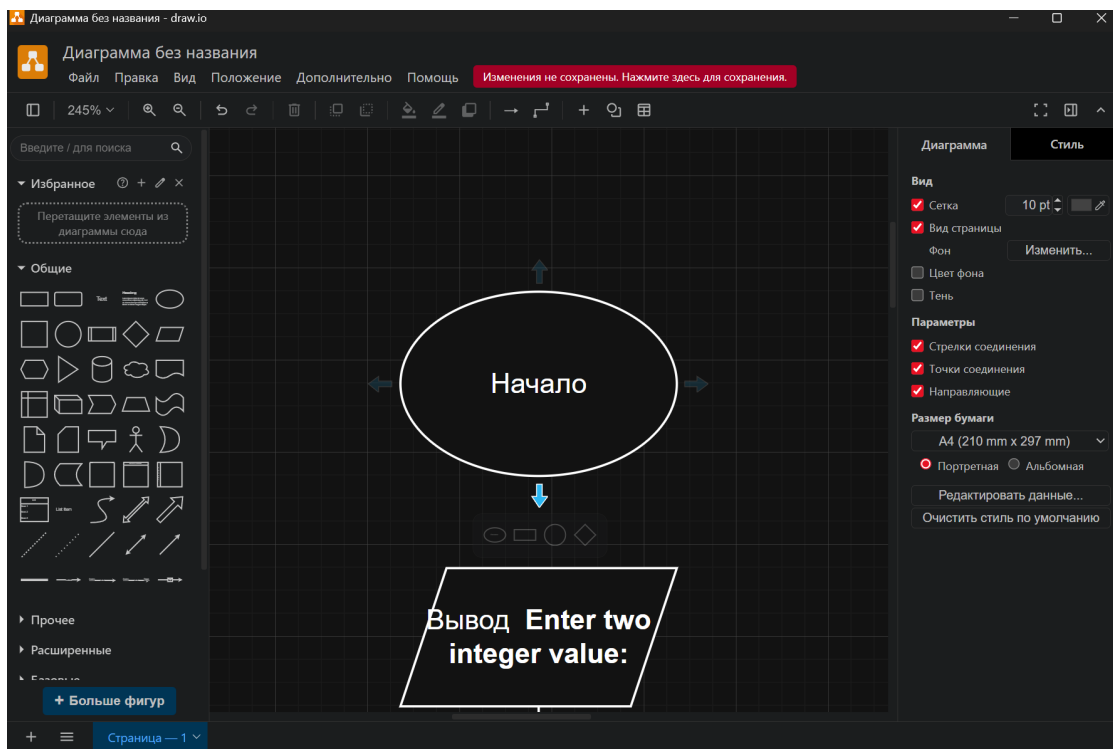


Рисунок 23 – Добавление соединительных линий.

Вывод: Таким образом, в ходе работы были сделана схема алгоритма и получены навыки работы со средой Draw.io.

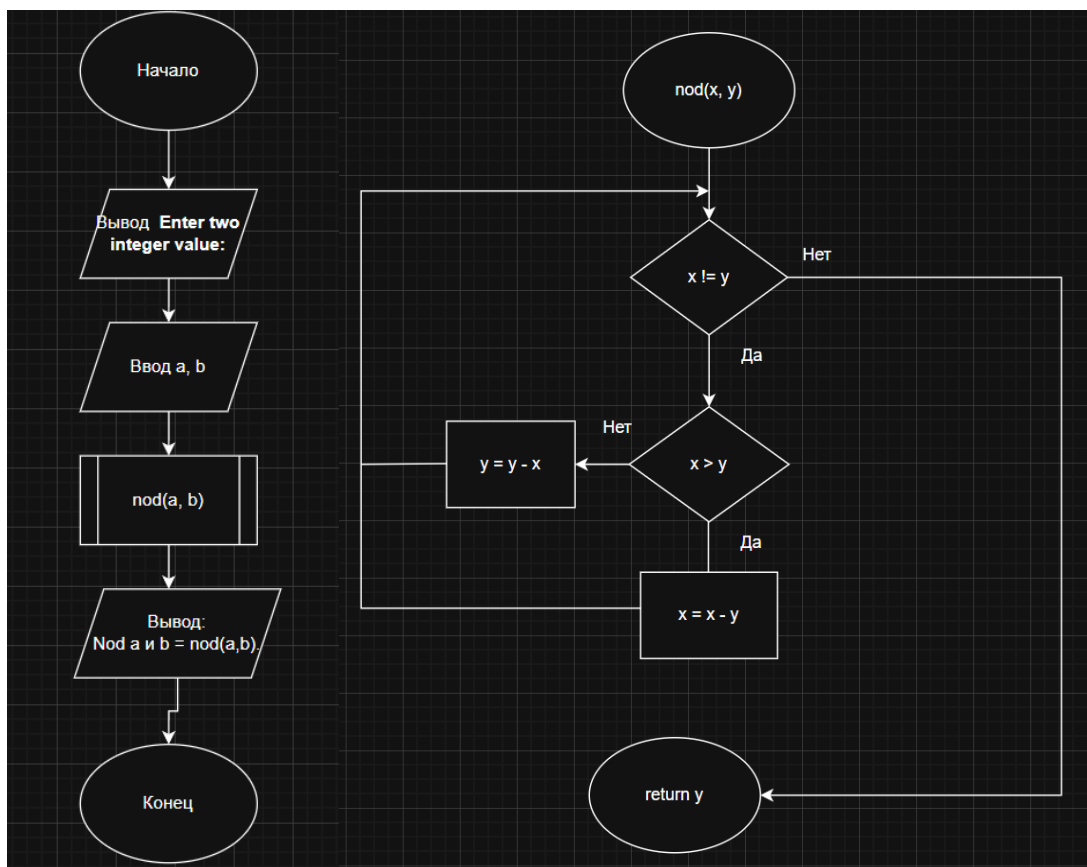


Рисунок 24 – Итоговые схемы.