Министерство транспорта Российской Федерации

Федеральное государственное автономное образовательное

учреждение высшего образования

«Российский университет транспорта»

(ФГАОУ ВО РУТ(МИИТ), РУТ (МИИТ)

Институт транспортной техники и систем управления

Кафедра «Управление и защита информации»

Лабораторная работа № 6

по дисциплине: «Программирование и основы алгоритмизации»

на тему: «Оператор переключения»

Выполнил: ст. гр. ТУУ-111

Богомолов В.Н.

Вариант №5

13.12.2024

(дата выполнения)

Проверил: к.т.н., доц. Сафронов А.И.

\_\_\_\_\_\_\_\_\_\_\_

(дата приёмки)

Москва – 2024 г.

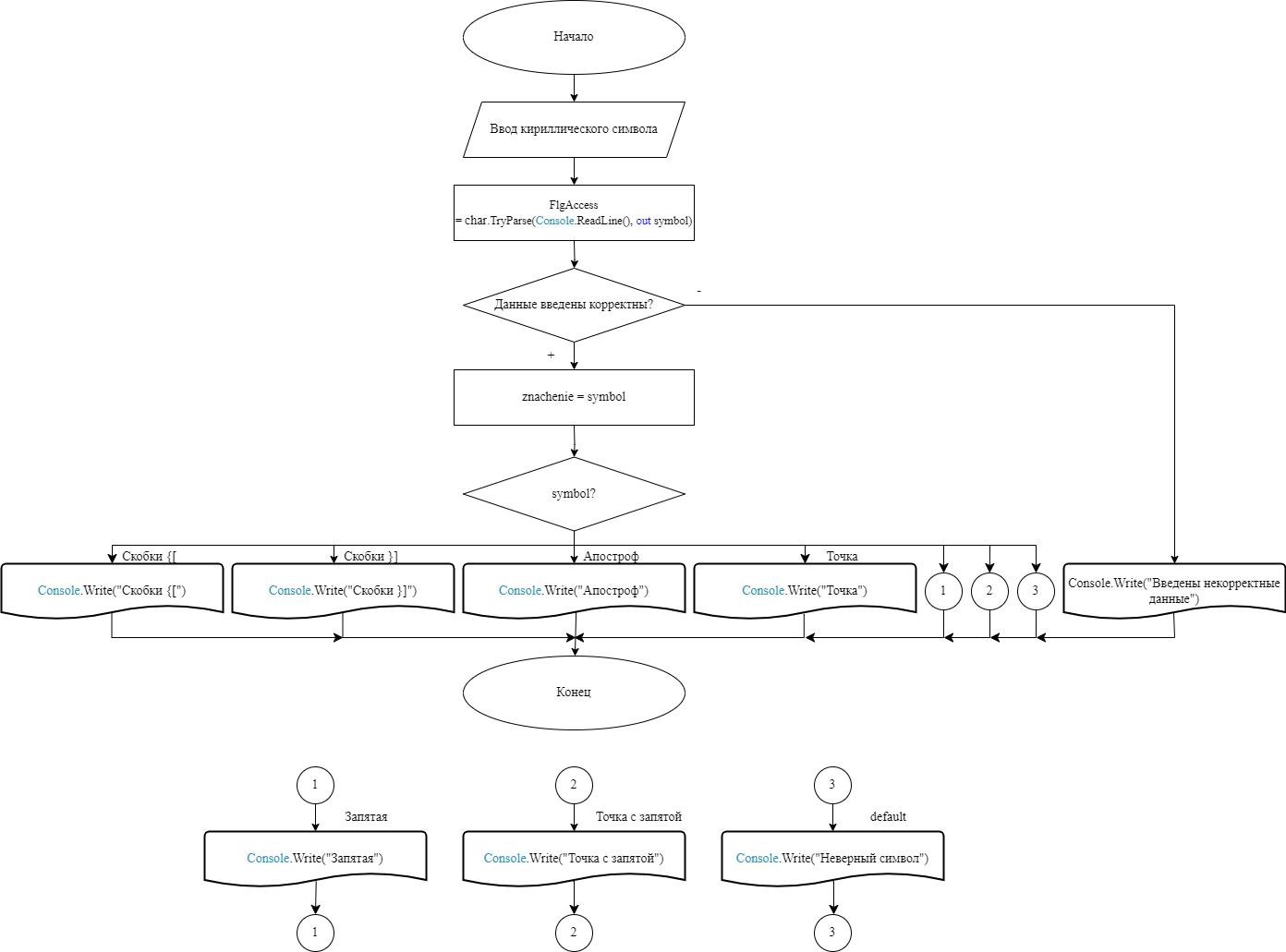
1. **Цель**

Решить индивидуальный вариант задания, используя условный оператор switch-case в интегрированной среде разработки (IDE) Microsoft Visual Studio на языке Visual C#. При этом запрещено использование условных операторов.

1. **Формулировка задачи**

В рамках данной работы необходимо реализовать алгоритм и программу, которая определяет знак препинания или операцию, ассоциируемую с введенным символом. При некорректном вводе должно выдаваться уведомление об ошибке. Задача должна быть решена с использованием оператора переключения и без условных операций.

1. **Блок-схема алгоритма**

****

3.1.1 – Блок-схема алгоритма

1. **Подбор тестовых примеров**

1) Вводим букву Х: на выводе мы получим скобки {[.

2) Вводим букву Ъ: на выводе мы получим скобки }].

3) Вводим букву Ж: на выводе мы получим точку с запятой ;.

4) Вводим букву Э: на выводе мы получим апостроф ‘.

5) Вводим букву Б: на выводе мы получим запятую ,.

6) Вводим букву Ю: на выводе мы получим точку .

7) Вводим букву Е : Некорректные данные, так как буква Е на клавиатуре не соответствует операция или знак препинания.

1. **Листинг кода**

using System; // Используем пространство имен System, которое содержит стандартные классы и функции

class Program

{

static void Main(string[] args) // Это главный метод программы, с которого начинается выполнение

{

char symbol; // Переменная для хранения символа, который введет пользователь

int znachenie; // Переменная для хранения числового значения символа

bool FlgAccess; // Переменная для проверки корректности ввода

// Выводим на экран приглашение для ввода символа

Console.WriteLine("Введите кирилический символ: ");

// Читаем ввод пользователя и пытаемся преобразовать его в символ

FlgAccess = char.TryParse(Console.ReadLine(), out symbol);

// Проверяем, успешно ли был введен символ

if (FlgAccess)

{

// Если ввод корректен, то сохраняем числовое значение символа в переменную znachenie

znachenie = symbol;

// Используем конструкцию switch для проверки символа и вывода соответствующего сообщения

switch (znachenie)

{

case 'х': // Если введен символ 'х'

Console.Write("Скобки {["); // Выводим текст

break; // Завершаем этот случай

case 'ъ': // Если введен символ 'ъ'

Console.Write("Скобки }]"); // Выводим текст

break;

case 'э': // Если введен символ 'э'

Console.Write("Апостроф"); // Выводим текст

break;

case 'ю': // Если введен символ 'ю'

Console.Write("Точка"); // Выводим текст

break;

case 'б': // Если введен символ 'б'

Console.Write("Запятая"); // Выводим текст

break;

case 'ж': // Если введен символ 'ж'

Console.Write("Точка с запятой"); // Выводим текст

break;

default: // Если введен символ, который не совпадает с любым из вышеуказанных

Console.Write("Неверный символ"); // Выводим сообщение об ошибке

break;

}

}

else

{

// Если введенные данные некорректны (например, если ввели не символ), выводим сообщение об ошибке

Console.Write("Введены некорректные данные");

}

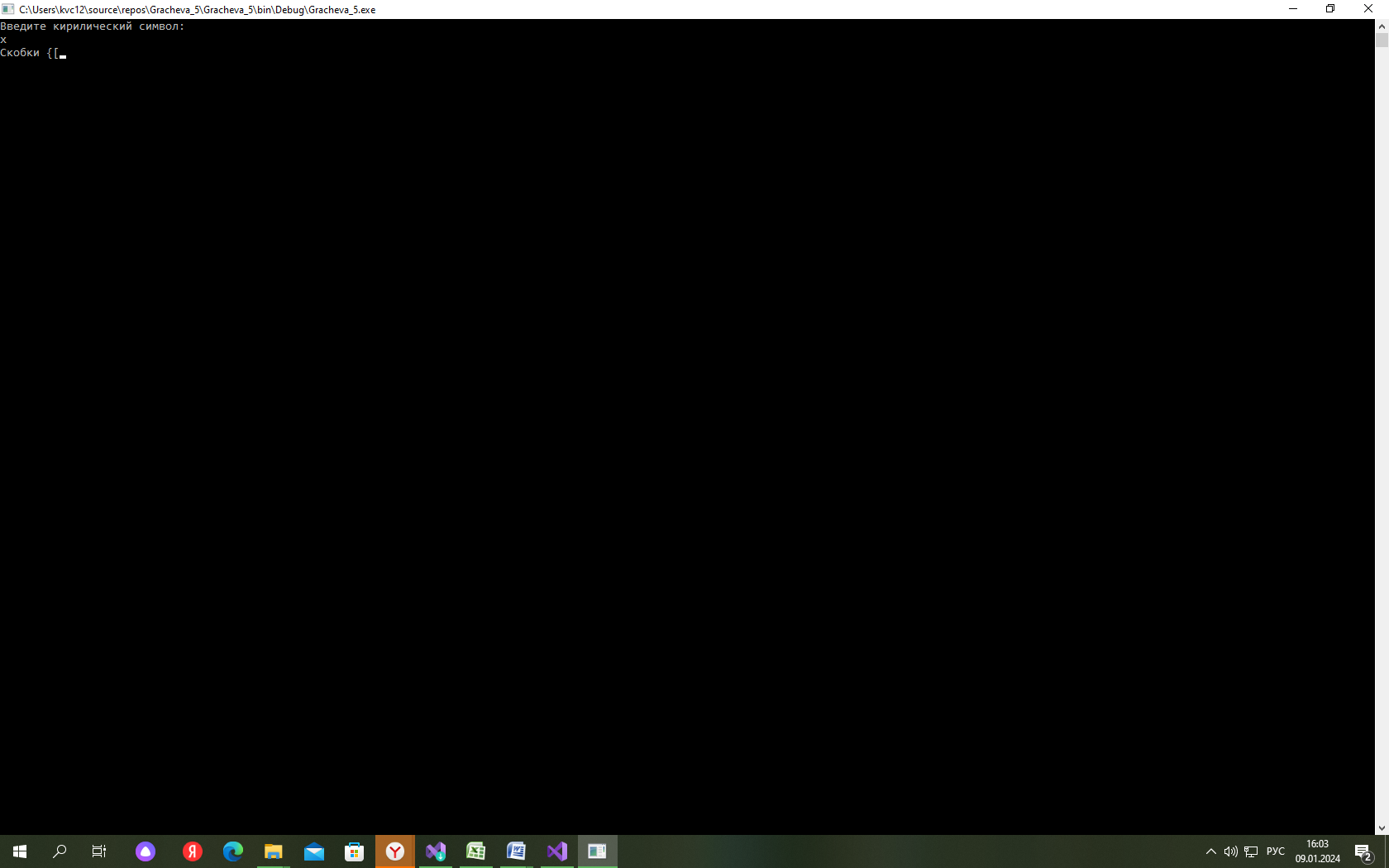
// Ожидаем нажатия клавиши перед завершением программы

Console.ReadKey(true);

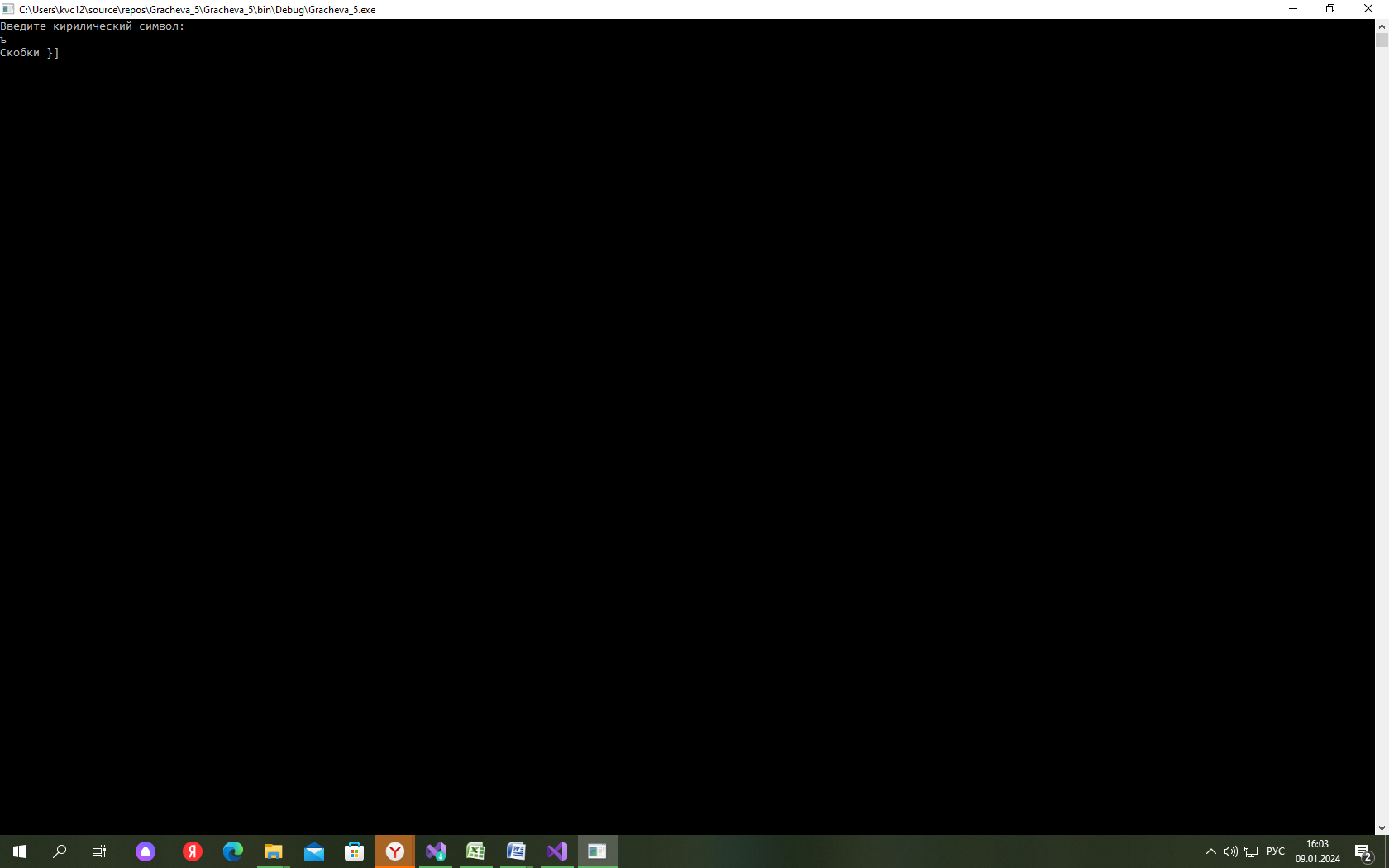
}

}

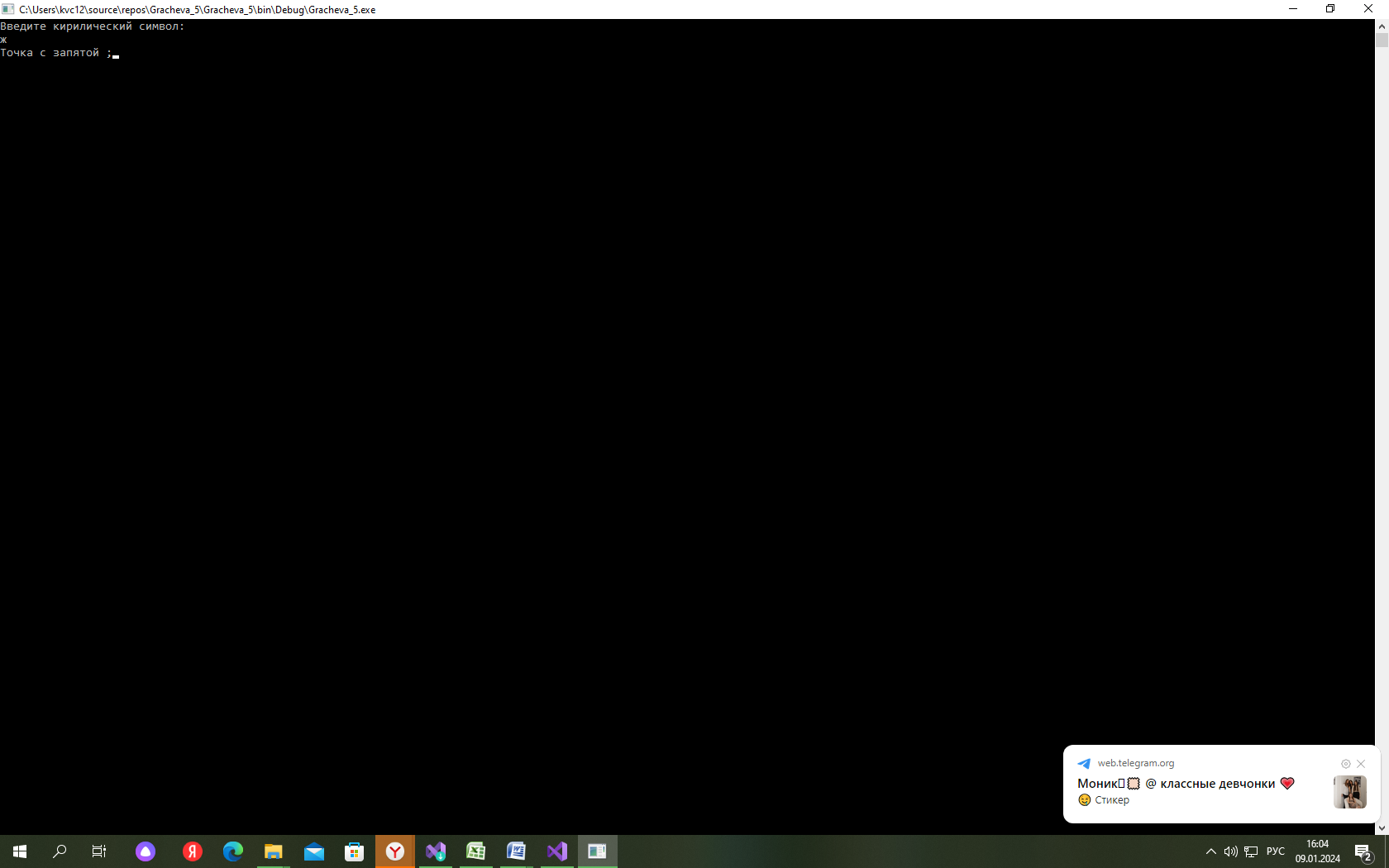
1. **Расчет тестовых примеров на ПК**

****

6.1.1 – Первый тестовый пример

****

6.1.2 – Второй тестовый пример

****

6.1.3 – Третий тестовый пример

1. **Вывод**

В результате выполнения работы была реализована задача, связанная с применением оператора переключения в программировании. Программа корректно идентифицирует введённые знаки препинания или выводит сообщение об ошибке при вводе неверного символа. Все знаки препинания и операции учитываются, а при вводе некорректных данных отображается сообщение "Ошибка! Некорректные данные". Тестирование показало, что алгоритм работает стабильно и соответствует поставленным требованиям. Данная работа способствует более глубокому пониманию принципов использования switch-case в языке C#.