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

Ванчаев.Д.С.

Вариант №10

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

(дата выполнения)

Проверил: к.т.н., доц. Сафронов А.И.

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

(дата приёмки)

Москва – 2024 г.

1. **Цель работы**

Решить индивидуальный вариант задания, используя условный оператор switch-case в интегрированной среде разработки (*IDE*) *Microsoft Visual Studio* на языке *Visual C#.*

1. **Формулировка задачи:**

Используя условный оператор switch-case, выполнить пользовательский ввод данных, который должен контролироваться, при помощи различных состояний метода “TryParse”. Решить индивидуальный вариант задания, учитывая ограничения, накладываемые не только на отдельные переменные, но и на участвующие в выражении функциональные зависимости, при заданной точности **.**

**Вариант №10**

Вводится римское число, соответствующее значению часа на циферблате. В ответ выдать направление / сторону света (учесть существование, например, «север-северо-запада», «юг-юго-востока» и т.д.).

1. **Схема алгоритма**

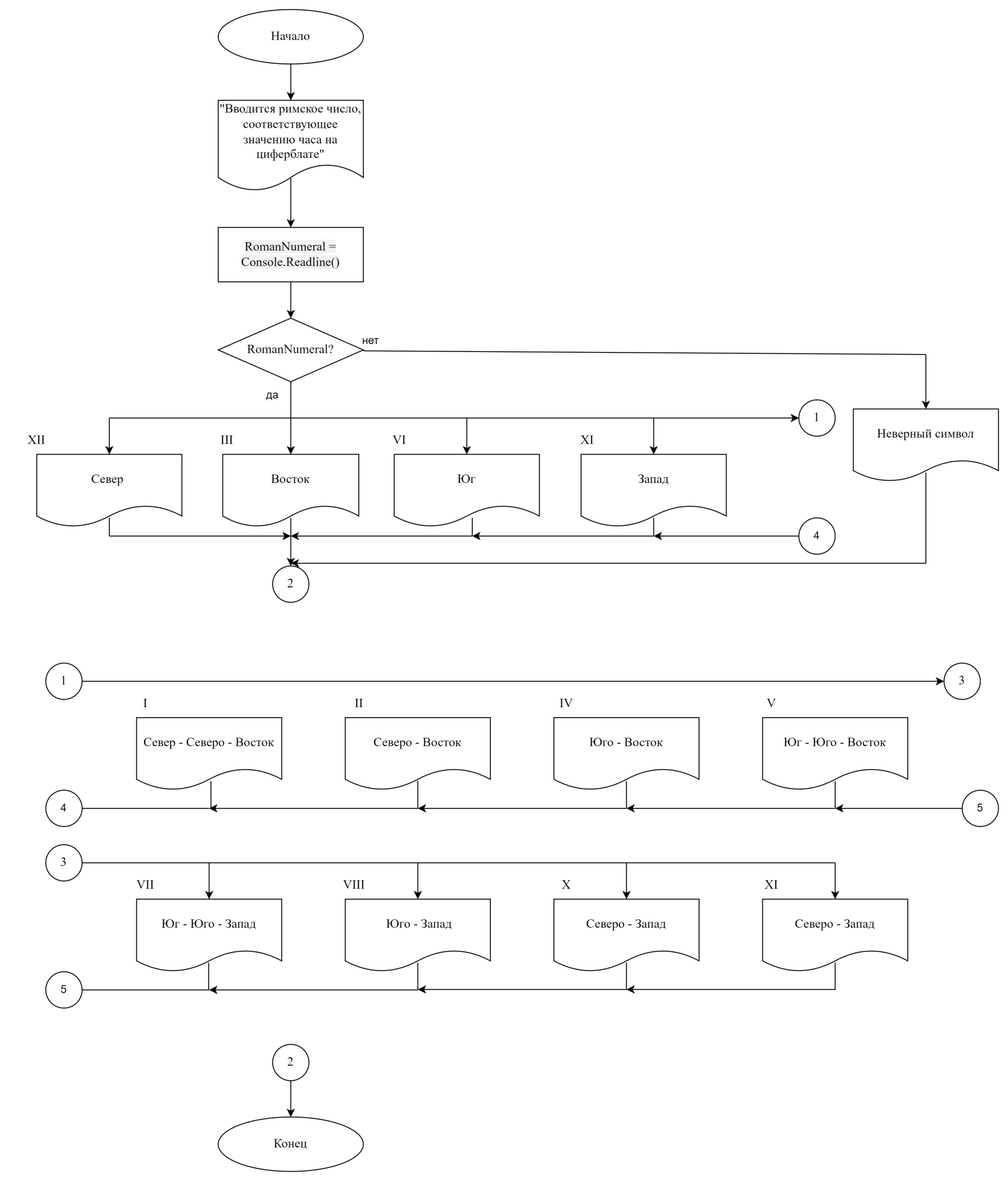


Рисунок 3.1 – блок схемы алгоритма

1. **Подбор тестовых примеров**
2. Вводим Римское число I: на выходе мы получим “Север-Северо-Восток”.
3. Вводим Римское число II: на выходе мы получим “Северо-Восток”.
4. Вводим Римское число III: на выходе мы получим “Восток”.
5. Вводим Римское число IV: на выходе мы получим “Юго-Восток”.
6. Вводим Римское число V: на выходе мы получим “Север-Северо-Запад”.
7. Вводим Римское число VI: на выходе мы получим “Юг”.
8. Вводим Римское число VII: на выходе мы получим “Юг-Юго-Восток”.
9. Вводим Римское число VIII: на выходе мы получим “Юг-Юго-Запад”.
10. Вводим Римское число IX: на выходе мы получим “Запад”.
11. Вводим Римское число X: на выходе мы получим “Юго-Запад”.
12. Вводим Римское число XI: на выходе мы получим “Северо-Запад”.
13. Вводим Римское число XII: на выходе мы получим “Север”.
14. Вводим Римское число XX: на выходе мы получим “Неверный символ”.
15. **Листинг**

using System;

using System.Diagnostics;

namespace Zadanie6

{

class Program

{

static void Main()

{

string RomanNumeral;

Console.WriteLine("Вводится римское число, соответствующее значению часа на циферблате:");

RomanNumeral = Console.ReadLine();

switch (RomanNumeral)

{

case "XII":

Console.Write("Север");

break;

case "III":

Console.Write("Восток");

break;

case "VI":

Console.Write("Юг");

break;

case "IX":

Console.Write("Запад");

break;

case "I":

Console.Write("Север-Северо-Восток");

break;

case "II":

Console.Write("Северо-Восток");

break;

case "IV":

Console.Write("Юго-Восток");

break;

case "V":

Console.Write("Юг-Юго-Восток");

break;

case "VII":

Console.Write("Юг-Юго-Запад");

break;

case "VIII":

Console.Write("Юго-Запад");

break;

case "X":

Console.Write("Северо-Запад");

break;

case "XI":

Console.Write("Север-Северо-Запад");

break;

default:

Console.Write("Неверный символ");

break;

}

Console.ReadKey(true);

}

}

}

1. **Расчёт тестовых примеров на ПК**

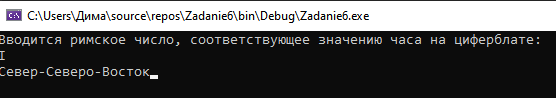


Рисунок 6.1 – расчёт тестового примера №1.

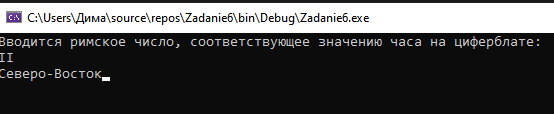


Рисунок 6.2 – расчёт тестового примера №2.

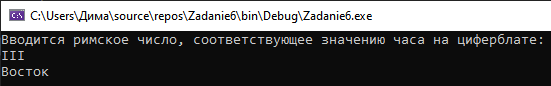


Рисунок 6.3 – расчёт тестового примера №3.

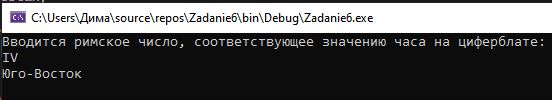


Рисунок 6.4 – расчёт тестового примера №4.

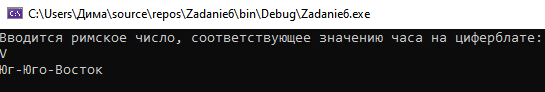


Рисунок 6.5 – расчёт тестового примера №5.

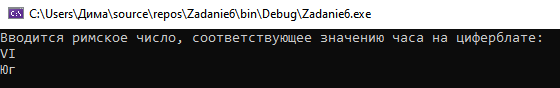


Рисунок 6.6 – расчёт тестового примера №6.

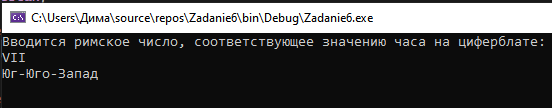


Рисунок 6.7 – расчёт тестового примера №7.

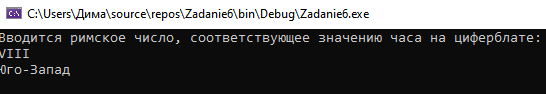


Рисунок 6.8 – расчёт тестового примера №8.

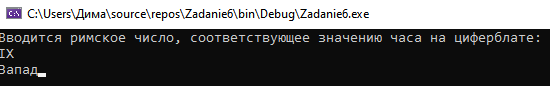


Рисунок 6.9 – расчёт тестового примера №9.

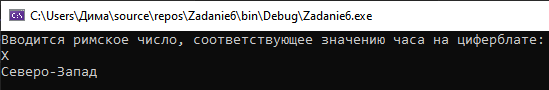


Рисунок 6.10 – расчёт тестового примера №10.

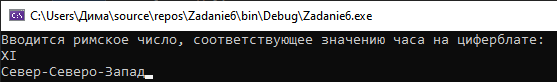


Рисунок 6.11 – расчёт тестового примера №11.

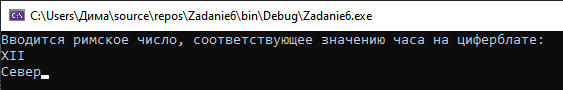


Рисунок 6.12 – расчёт тестового примера №12.

**7. Вывод по работе**

Был решен индивидуальный вариант задания, используя условный оператор switch-case в интегрированной среде разработки (*IDE*) *Microsoft Visual Studio* на языке *Visual C#.*