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 № 17

по дисциплине: «Программирование и основы алгоритмизации»

на тему: «Автоматизация Word»

Выполнил: ст. гр. ТУУ-211

Волобуев Е.Д.

Вариант №2

16.12.2024

(дата выполнения)

Проверил: к.т.н., доц. Сафронов А.И.

16.12.2024

(дата приёмки)

Москва – 2024 г.

**Оглавление**

[1. Цель работы 3](#_Toc184607077)

[2. Формулировка задачи 3](#_Toc184607078)

[3. Сеть Петри 4](#_Toc184607079)

[4. Составление схем алгоритмов методов в составе решения 4](#_Toc184607080)

[5. Подбор тестовых примеров 7](#_Toc184607081)

[6. Листинг (код) составленного программного обеспечения 9](#_Toc184607082)

[7. Графический пользовательский интерфейс программного обеспечения и его описание 13](#_Toc184607083)

[8. Подтверждение соответствия графического пользовательского интерфейса требованиям к оформлению 16](#_Toc184607084)

[**9. Расчёт тестовых примеров с использованием составленного программного обеспечения** 18](#_Toc184607085)

[10. Формулировку вывода о проделанной работе 20](#_Toc184607086)

# **1. Цель работы**

Освоить навыки подключения внешних модулей (библиотек классов) и их использования в разрабатываемых программах; освоить навыки программируемой настройки параметров документов *Microsoft Office Word*.

# **2. Формулировка задачи**

В интегрированной среде разработки *Microsoft Visual Studio* составить программу в режиме *Windows Forms Application* на языке *Visual C#*, представляющую собой средство взаимодействия с текстовым файлом, необходимым для настройки формата надписей, заносимых в документ *Microsoft Office Word.* Создать и настроить графический пользовательский интерфейс, взаимодействующий с модулем (библиотекой классов) текстового редактора *Microsoft Office Word*, предназначенный для создания двух документов: титульного листа и индивидуального документа. Должна быть предусмотрена кнопка «Создать титульный лист», инициирующая генерацию титульного листа в формате *Microsoft Office Word.* Индивидуальный документ, представленный иллюстрацией, должен быть первично набран в *Microsoft Office Word* вручную. Он запланирован в качестве тестового примера в данной работе. Должна быть предусмотрена кнопка «Создать документ», инициирующая генерацию индивидуального документа в формате *Microsoft Office Word.* Должен быть предусмотрен предварительный просмотр документа, планируемого к выгрузке. Все заполнители в работе выполняются при помощи «кареток» [, , ] (*Tab Stops*) и табуляции [https://prosto-kod.ru/17_01_2023/4.jpg] (*Tab*). Все неявные колоночные выравнивания допустимо выполнять необрамлёнными таблицами (таблицами без визуальных границ). Предоставить пользователю возможность для выгрузки «по галочке» не только титульного листа, но и полноценного шаблона для отчёта по работе с разделами: «Цель работы», «Задача», «Содержательная часть», «Вывод». Каждый раздел должен располагаться на отдельной странице. Использовать для переноса не пустые абзацы, а разрывы страницы.

# **3. Сеть Петри**

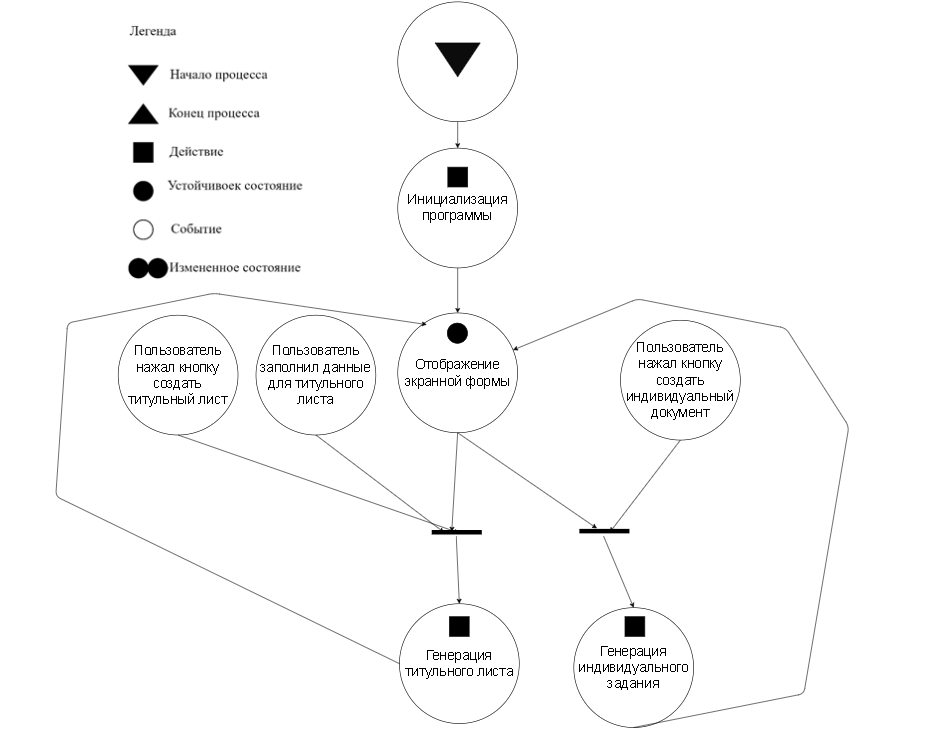


Рисунок 1 – Сеть Петри

# **4. Составление схем алгоритмов методов в составе решения**

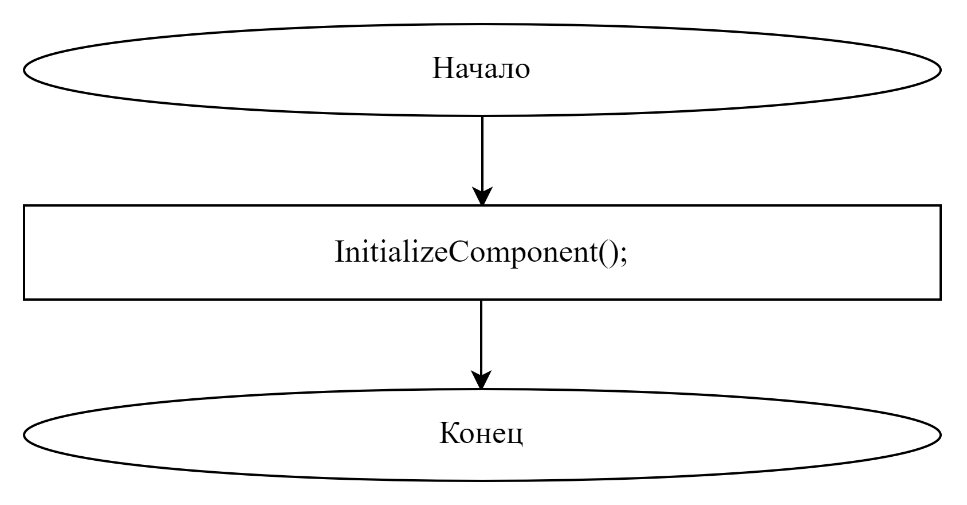


Рисунок 2 – Инициализация программы

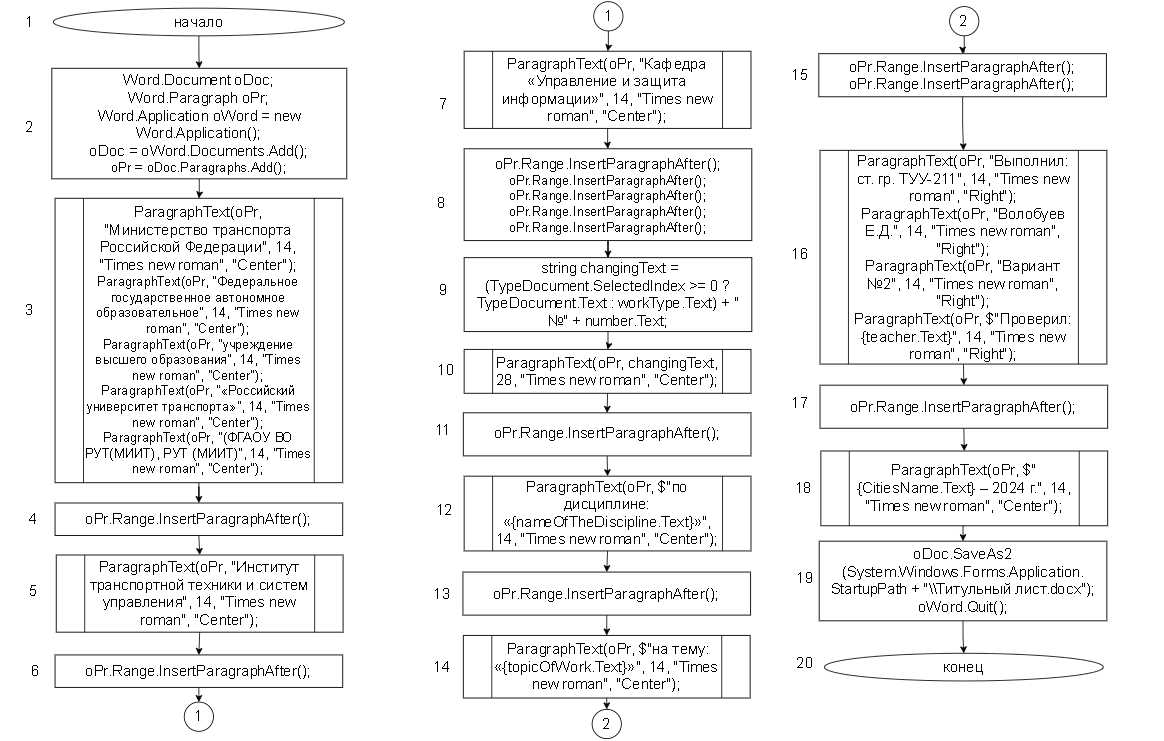


Рисунок 3 – Блок-схема метода *CreateATitle\_Click*

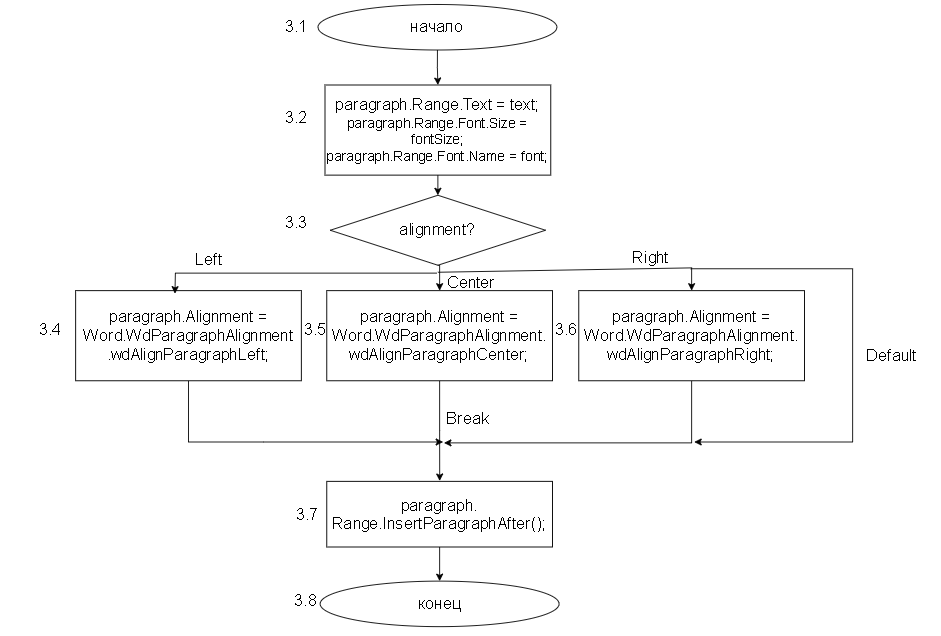


Рисунок 4 – Блок-схема метода *ParagraphText*

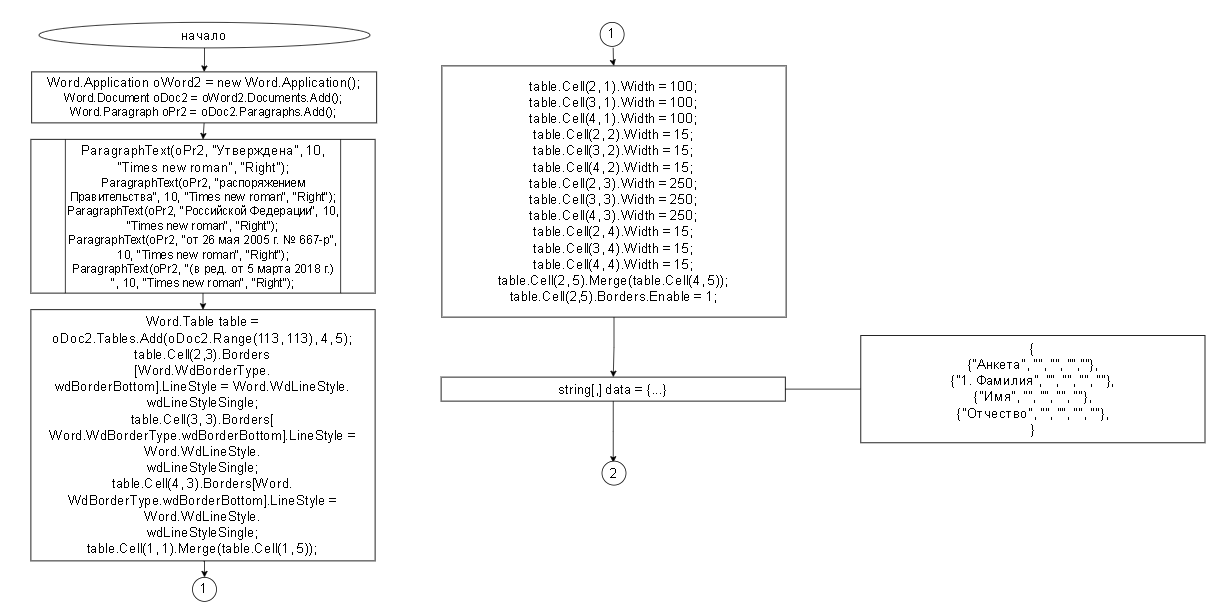


Рисунок 5 – Блок-схема метода *CreateAIndividualDocument\_Click*

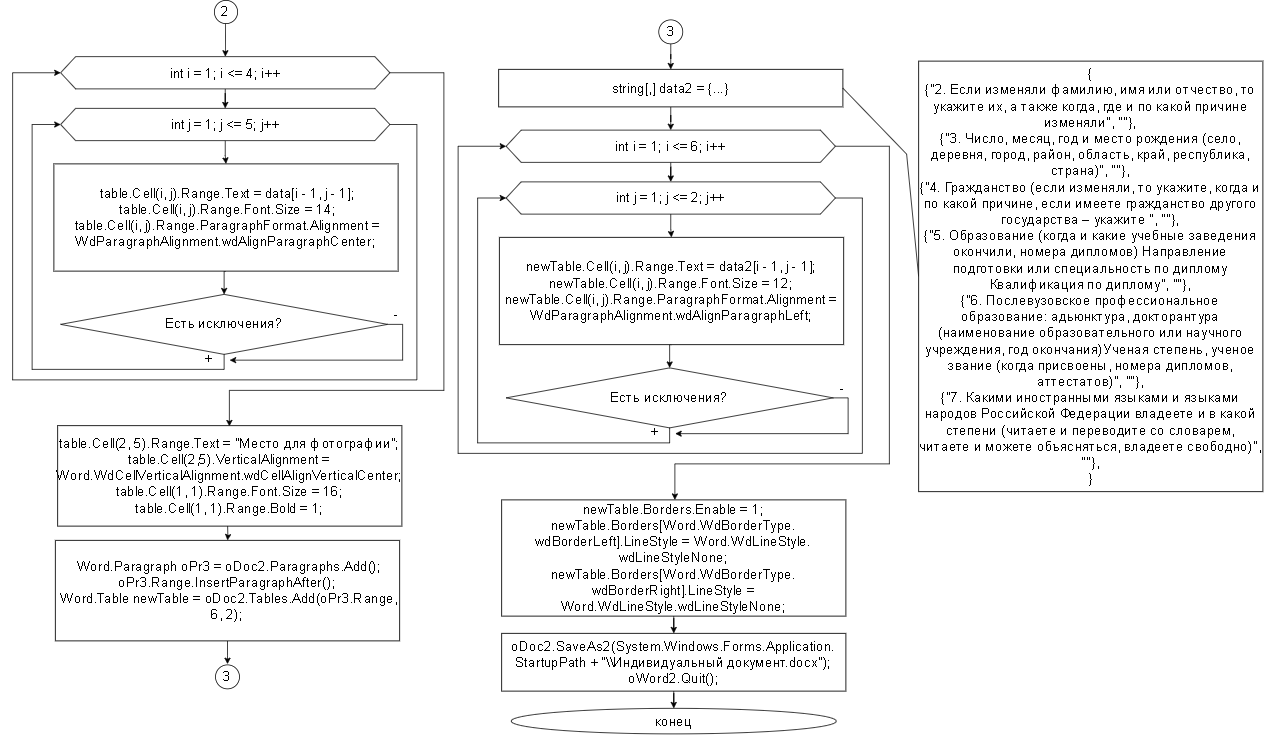


Рисунок 6 – Блок-схема метода *CreateAIndividualDocument\_Click*

# **5. Подбор тестовых примеров**

*Тестовый пример общей части задания:*

Вид отчетного документа: Реферат

Номер работы: 6

Тема работы: Автоматизация Word

Наименование дисциплины: Программирование и основы алгоритмизации

Преподаватель: Сафронов А.И.

*Тестовый пример индивидуальной части задания:*

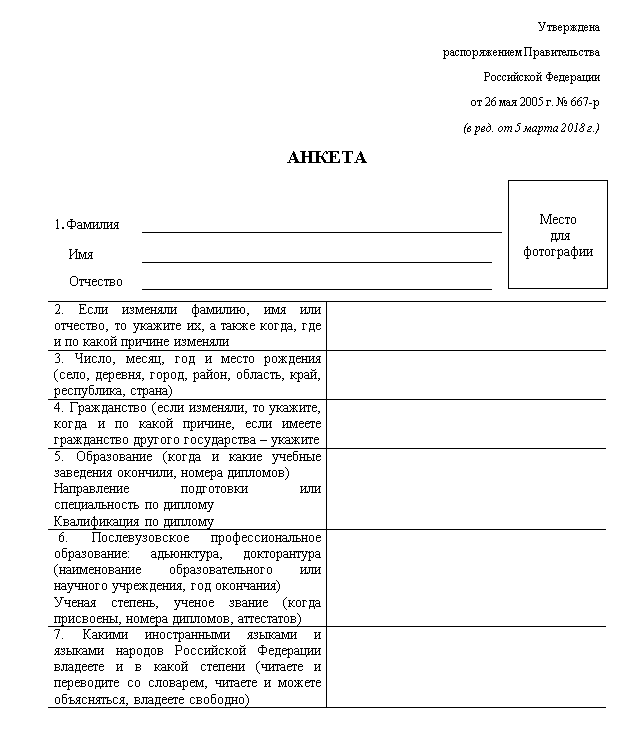


Рисунок 7 – Тестовый пример индивидуальной части задания

# **6. Листинг (код) составленного программного обеспечения**

using System;

using System.Windows.Forms;

using Word = Microsoft.Office.Interop.Word;

using System.Reflection;

using Microsoft.Office.Interop.Word;

namespace Rabota\_17

{

public partial class WordApp : Form

{

private Word.Application wordApp;

private Word.Document docTwo;

object ObjMissing = Missing.Value;

private string[] typeDocument = { "Отчёт", "Реферат", "Эссе", "Курсовой проект",

"Курсовая работа", "Доклад", "Домашнее задание"};

private string[] typeOfWork = { "Лабораторная работа", "Практическая работа", "Индивидуальное задание",

"Учебная практика", "Производственная практика", "Преддипломная практика"};

private string[] numberOfWork = { "1", "2", "3", "4", "5", "6", "7", "8", "9", "10" };

private string[] Cities = { "Москва", "Санкт-Петербург", "Казань", "Орёл", "Чебоксары", "Тверь", "Старый Оскол", "Нижний Новгород" };

public WordApp()

{

InitializeComponent();

TypeDocument.Items.AddRange(typeDocument);

workType.Items.AddRange(typeOfWork);

number.Items.AddRange(numberOfWork);

CitiesName.Items.AddRange(Cities);

}

private static void ParagraphText(Paragraph paragraph, string text, int fontSize, string font, string alignment)

{

paragraph.Range.Text = text;

paragraph.Range.Font.Size = fontSize;

paragraph.Range.Font.Name = font;

switch (alignment)

{

case "Right":

paragraph.Alignment = Word.WdParagraphAlignment.wdAlignParagraphRight;

break;

case "Left":

paragraph.Alignment = Word.WdParagraphAlignment.wdAlignParagraphLeft;

break;

case "Center":

paragraph.Alignment = Word.WdParagraphAlignment.wdAlignParagraphCenter;

break;

default:

break;

}

paragraph.Range.InsertParagraphAfter();

}

private void CreateATitle\_Click(object sender, EventArgs e)

{

Word.Document oDoc;

Word.Paragraph oPr;

Word.Application oWord = new Word.Application();

oDoc = oWord.Documents.Add();

oPr = oDoc.Paragraphs.Add();

ParagraphText(oPr, "Министерство транспорта Российской Федерации", 14, "Times new roman", "Center");

ParagraphText(oPr, "Федеральное государственное автономное образовательное", 14, "Times new roman", "Center");

ParagraphText(oPr, "учреждение высшего образования", 14, "Times new roman", "Center");

ParagraphText(oPr, "«Российский университет транспорта»", 14, "Times new roman", "Center");

ParagraphText(oPr, "(ФГАОУ ВО РУТ(МИИТ), РУТ (МИИТ)", 14, "Times new roman", "Center");

oPr.Range.InsertParagraphAfter();

ParagraphText(oPr, "Институт транспортной техники и систем управления", 14, "Times new roman", "Center");

oPr.Range.InsertParagraphAfter();

ParagraphText(oPr, "Кафедра «Управление и защита информации»", 14, "Times new roman", "Center");

oPr.Range.InsertParagraphAfter();

oPr.Range.InsertParagraphAfter();

oPr.Range.InsertParagraphAfter();

oPr.Range.InsertParagraphAfter();

oPr.Range.InsertParagraphAfter();

string changingText = (TypeDocument.SelectedIndex >= 0 ? TypeDocument.Text : workType.Text) + " №" + number.Text;

ParagraphText(oPr, changingText, 28, "Times new roman", "Center");

oPr.Range.InsertParagraphAfter();

ParagraphText(oPr, $"по дисциплине: «{nameOfTheDiscipline.Text}»", 14, "Times new roman", "Center");

oPr.Range.InsertParagraphAfter();

ParagraphText(oPr, $"на тему: «{topicOfWork.Text}»", 14, "Times new roman", "Center");

oPr.Range.InsertParagraphAfter();

oPr.Range.InsertParagraphAfter();

ParagraphText(oPr, "Выполнил: ст. гр. ТУУ-211", 14, "Times new roman", "Right");

ParagraphText(oPr, "Волобуев Е.Д.", 14, "Times new roman", "Right");

ParagraphText(oPr, "Вариант №2", 14, "Times new roman", "Right");

ParagraphText(oPr, $"Проверил: {teacher.Text}", 14, "Times new roman", "Right");

oPr.Range.InsertParagraphAfter();

ParagraphText(oPr, $"{CitiesName.Text} – 2024 г.", 14, "Times new roman", "Center");

oDoc.SaveAs2(System.Windows.Forms.Application.StartupPath + "\\Титульный лист.docx");

oWord.Quit();

}

private void CreateAIndividualDocument\_Click(object sender, EventArgs e)

{

Word.Application oWord2 = new Word.Application();

Word.Document oDoc2 = oWord2.Documents.Add();

Word.Paragraph oPr2 = oDoc2.Paragraphs.Add();

ParagraphText(oPr2, "Утверждена", 10, "Times new roman", "Right");

ParagraphText(oPr2, "распоряжением Правительства", 10, "Times new roman", "Right");

ParagraphText(oPr2, "Российской Федерации", 10, "Times new roman", "Right");

ParagraphText(oPr2, "от 26 мая 2005 г. № 667-р", 10, "Times new roman", "Right");

ParagraphText(oPr2, "(в ред. от 5 марта 2018 г.) ", 10, "Times new roman", "Right");

Word.Table table = oDoc2.Tables.Add(oDoc2.Range(113, 113), 4, 5);

table.Cell(2,3).Borders[Word.WdBorderType.wdBorderBottom].LineStyle = Word.WdLineStyle.wdLineStyleSingle;

table.Cell(3, 3).Borders[Word.WdBorderType.wdBorderBottom].LineStyle = Word.WdLineStyle.wdLineStyleSingle;

table.Cell(4, 3).Borders[Word.WdBorderType.wdBorderBottom].LineStyle = Word.WdLineStyle.wdLineStyleSingle;

table.Cell(1, 1).Merge(table.Cell(1, 5));

table.Cell(2, 1).Width = 100;

table.Cell(3, 1).Width = 100;

table.Cell(4, 1).Width = 100;

table.Cell(2, 2).Width = 15;

table.Cell(3, 2).Width = 15;

table.Cell(4, 2).Width = 15;

table.Cell(2, 3).Width = 250;

table.Cell(3, 3).Width = 250;

table.Cell(4, 3).Width = 250;

table.Cell(2, 4).Width = 15;

table.Cell(3, 4).Width = 15;

table.Cell(4, 4).Width = 15;

table.Cell(2, 5).Merge(table.Cell(4, 5));

table.Cell(2,5).Borders.Enable = 1;

string[,] data = {

{"Анкета", "", "", "",""},

{"1. Фамилия", "", "", "", ""},

{"Имя", "", "", "", ""},

{"Отчество", "", "", "", ""},

};

for (int i = 1; i <= 4; i++)

{

for (int j = 1; j <= 5; j++)

{

try

{

table.Cell(i, j).Range.Text = data[i - 1, j - 1];

table.Cell(i, j).Range.Font.Size = 14;

table.Cell(i, j).Range.ParagraphFormat.Alignment = WdParagraphAlignment.wdAlignParagraphCenter;

}

catch

{

}

}

}

table.Cell(2, 5).Range.Text = "Место для фотографии";

table.Cell(2,5).VerticalAlignment = Word.WdCellVerticalAlignment.wdCellAlignVerticalCenter;

table.Cell(1, 1).Range.Font.Size = 16;

table.Cell(1, 1).Range.Bold = 1;

Word.Paragraph oPr3 = oDoc2.Paragraphs.Add();

oPr3.Range.InsertParagraphAfter();

Word.Table newTable = oDoc2.Tables.Add(oPr3.Range, 6, 2);

string[,] data2 = {

{"2. Если изменяли фамилию, имя или отчество, то укажите их, а также когда, где и по какой причине изменяли", ""},

{"3. Число, месяц, год и место рождения (село, деревня, город, район, область, край, республика, страна)", ""},

{"4. Гражданство (если изменяли, то укажите, когда и по какой причине, если имеете гражданство другого государства – укажите ", ""},

{"5. Образование (когда и какие учебные заведения окончили, номера дипломов) Направление подготовки или специальность по диплому Квалификация по диплому", ""},

{"6. Послевузовское профессиональное образование: адьюнктура, докторантура (наименование образовательного или научного учреждения, год окончания)Ученая степень, ученое звание (когда присвоены, номера дипломов, аттестатов)", ""},

{"7. Какими иностранными языками и языками народов Российской Федерации владеете и в какой степени (читаете и переводите со словарем, читаете и можете объясняться, владеете свободно)", ""},

};

for (int i = 1; i <= 6; i++)

{

for (int j = 1; j <= 2; j++)

{

try

{

newTable.Cell(i, j).Range.Text = data2[i - 1, j - 1];

newTable.Cell(i, j).Range.Font.Size = 12;

newTable.Cell(i, j).Range.ParagraphFormat.Alignment = WdParagraphAlignment.wdAlignParagraphLeft;

}

catch

{

}

}

}

newTable.Borders.Enable = 1;

newTable.Borders[Word.WdBorderType.wdBorderLeft].LineStyle = Word.WdLineStyle.wdLineStyleNone;

newTable.Borders[Word.WdBorderType.wdBorderRight].LineStyle = Word.WdLineStyle.wdLineStyleNone;

oDoc2.SaveAs2(System.Windows.Forms.Application.StartupPath + "\\Индивидуальный документ.docx");

oWord2.Quit();

}

}

}

# **7. Графический пользовательский интерфейс программного обеспечения и его описание**

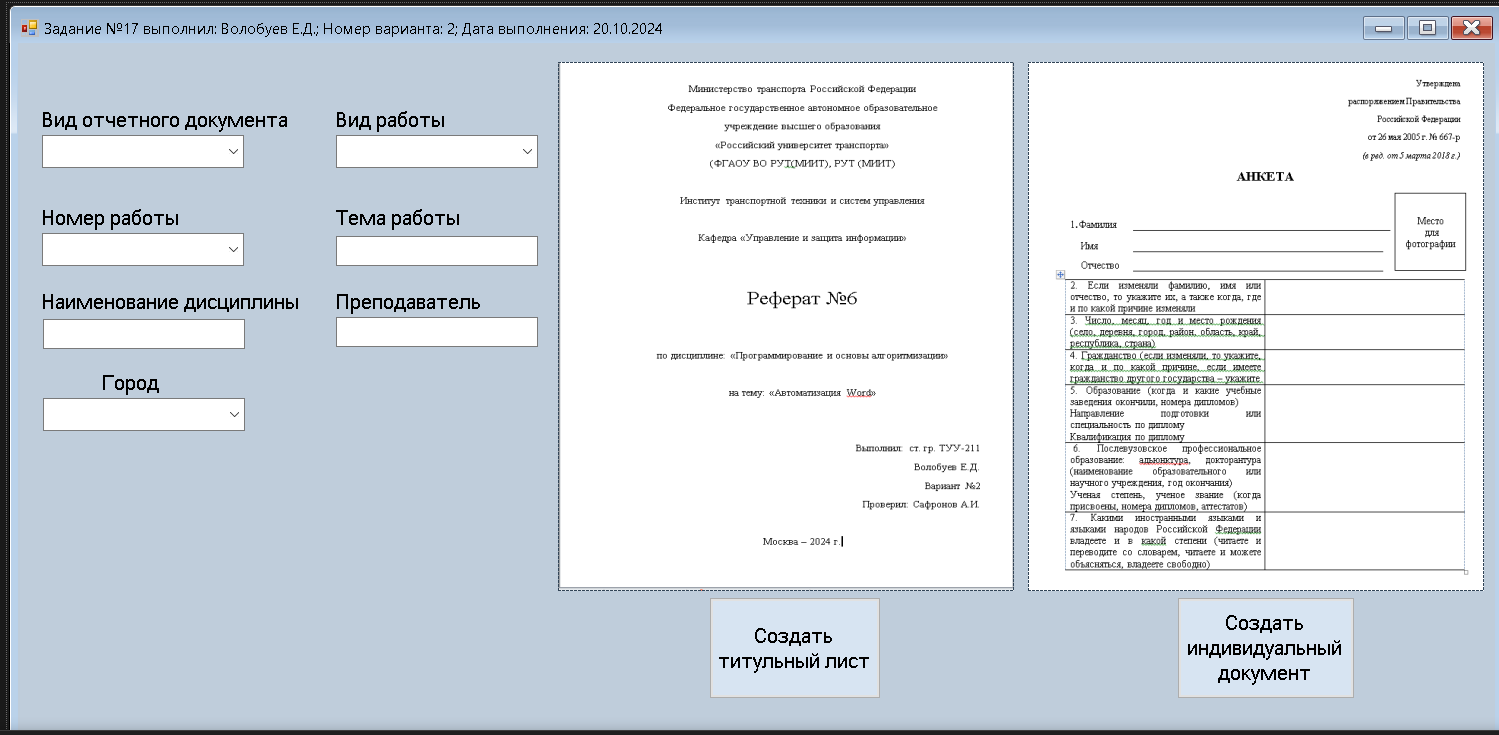


Рисунок 8 – Графический пользовательский интерфейс программного обеспечения

**Требования к оформлению графического пользовательского интерфейса**:

1. Заголовок главной формы должен содержать надпись вида: «Задание №5 выполнил: [Фамилия И.О. автора]; Номер варианта: [Номер]; Дата выполнения: [дд/мм/гггг]».

2. Дата выполнения проставляется в момент, когда программа считается законченной и по ней можно готовить итоговый отчёт о выполнении работы. Этот момент должен быть согласован с преподавателем, ведущим учебную дисциплину.

3. Имя автора-разработчика графического пользовательского интерфейса вводится через свойства (*Properties*) экранной формы и не должно быть подвержено изменению/редактированию в коде. Это же имя должно быть использовано для вставки в строку Ф.И.О. автора-составителя на титульном листе.

4. Имя и должность лица, проверяющего работу, вводится либо через текстовое поле (*Text Box*), либо через комбинированный список (*Combo Box*), если условием индивидуального задания по варианту не предусмотрено иное.

5. Предусмотреть на графическом пользовательском интерфейсе комбинированный список (*Combo Box*), в котором выбирается вид отчётного документа:

- отчёт;

- реферат;

- эссе;

- курсовой проект;

- курсовая работа;

- доклад;

- домашнее задание.

6. Предусмотреть на графическом пользовательском интерфейсе комбинированный список (*Combo Box*), в котором выбирается вид работы:

- лабораторная работа;

- практическая работа;

- индивидуальное задание;

- учебная практика;

- производственная практика;

- преддипломная практика.

7. Предусмотреть комбинированный список (*Combo Box*), позволяющий выбрать номер работы (диапазон от 1 до 10).

8. Предусмотреть текстовое поле для ввода (*Text Box*) темы работы.

9. Предусмотреть текстовое поле для ввода (*Text Box*) наименования дисциплины, в рамках которой выполняется работа.

10. Предусмотреть текстовое поле для ввода (*Text Box*) города – выполнено.

# **8. Подтверждение соответствия графического пользовательского интерфейса требованиям к оформлению**

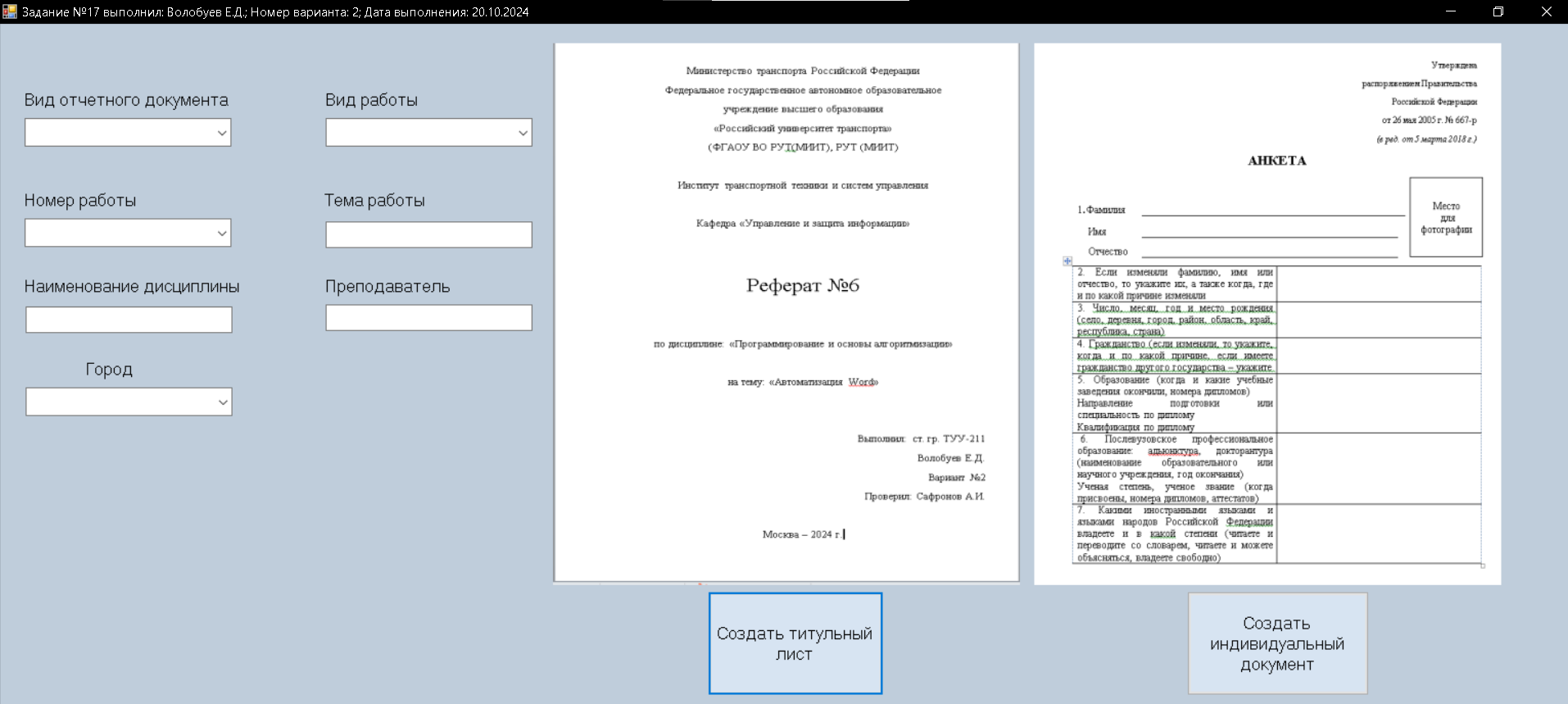


Рисунок 9 – Подтверждение соответствия графического пользовательского интерфейса требованиям к оформлению

**Выполненные** требования к оформлению графического пользовательского интерфейса:

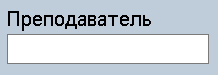
1. Заголовок главной формы должен содержать надпись вида: «Задание №5 выполнил: [Фамилия И.О. автора]; Номер варианта: [Номер]; Дата выполнения: [дд/мм/гггг]» – выполнено.



2. Дата выполнения проставляется в момент, когда программа считается законченной и по ней можно готовить итоговый отчёт о выполнении работы. Этот момент должен быть согласован с преподавателем, ведущим учебную дисциплину – выполнено.

3. Имя автора-разработчика графического пользовательского интерфейса вводится через свойства (*Properties*) экранной формы и не должно быть подвержено изменению/редактированию в коде. Это же имя должно быть использовано для вставки в строку Ф.И.О. автора-составителя на титульном листе – выполнено.

4. Имя и должность лица, проверяющего работу, вводится либо через текстовое поле (*Text Box*), либо через комбинированный список (*Combo Box*), если условием индивидуального задания по варианту не предусмотрено иное – выполнено.



5. Предусмотрел на графическом пользовательском интерфейсе комбинированный список (*Combo Box*), в котором выбирается вид отчётного документа:

- отчёт;

- реферат;

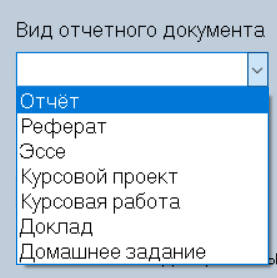
- эссе;

- курсовой проект;

- курсовая работа;

- доклад;

- домашнее задание.



6. Предусмотрел на графическом пользовательском интерфейсе комбинированный список (*Combo Box*), в котором выбирается вид работы:

- лабораторная работа;

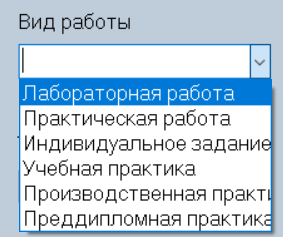
- практическая работа;

- индивидуальное задание;

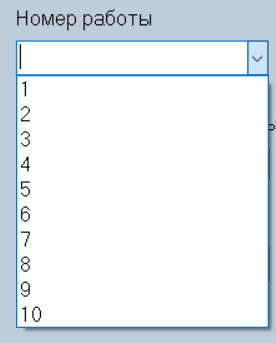
- учебная практика;

- производственная практика;

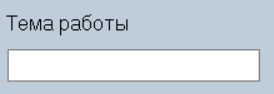
- преддипломная практика.



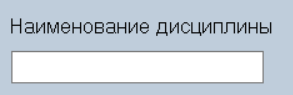
7. Предусмотреть комбинированный список (*Combo Box*), позволяющий выбрать номер работы (диапазон от 1 до 10) – выполнено



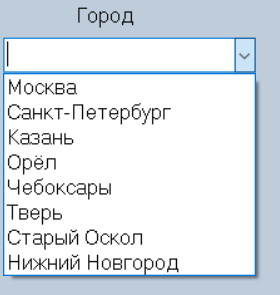
8. Предусмотреть текстовое поле для ввода (*Text Box*) темы работы – выполнено.



9. Предусмотреть текстовое поле для ввода (*Text Box*) наименования дисциплины, в рамках которой выполняется работа – выполнено.



10. Предусмотреть текстовое поле для ввода (*Text Box*) города – выполнено.



## **9. Расчёт тестовых примеров с использованием составленного программного обеспечения**

*Тестирование примера для общей части задания:*

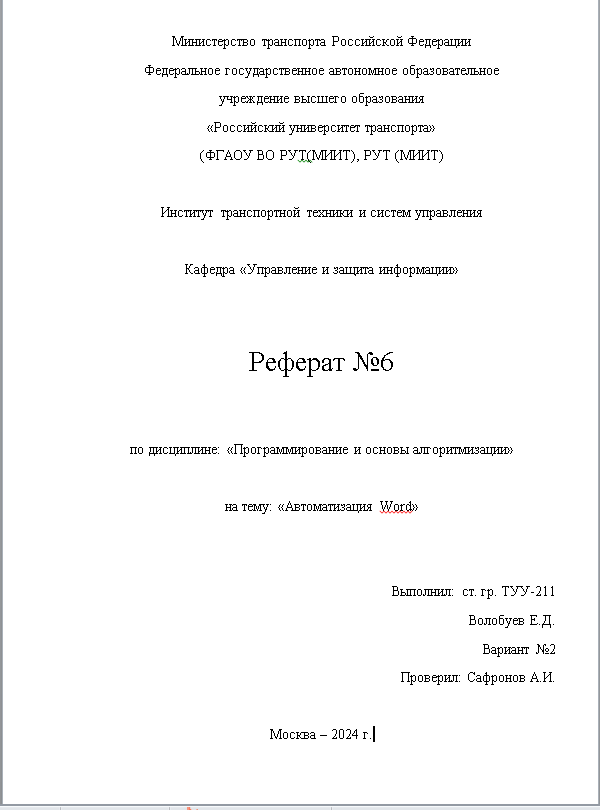


Рисунок 10 – Расчёт тестового примера для общей части задания

*Тестирование индивидуальной части задания:*

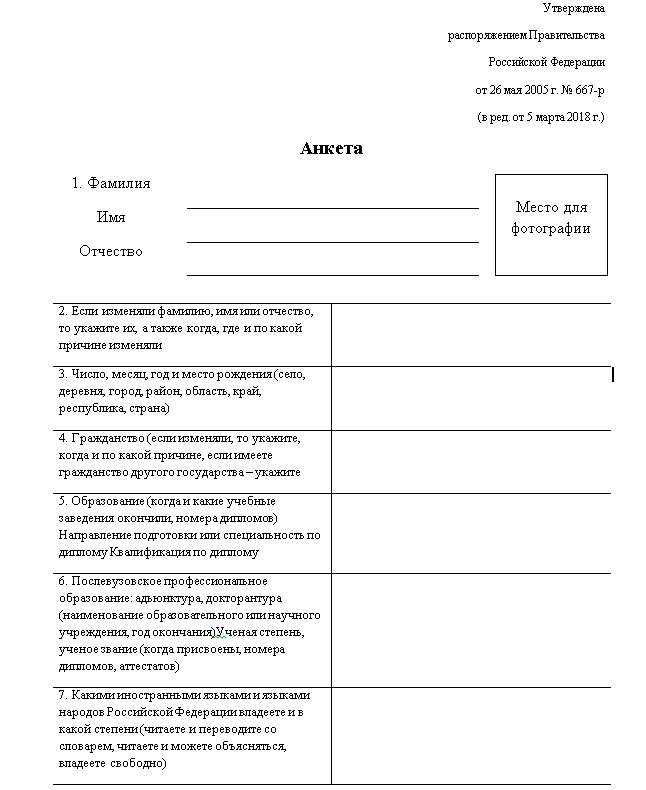


Рисунок 11 – Расчёт тестового примера для индивидуальной части

# **10. Формулировку вывода о проделанной работе**

В процессе выполнения задания значительно расширил свои навыки и знания в области работы с внешними библиотеками, что позволило мне успешно интегрировать дополнительные модули в свои проекты. Освоение методов подключения и использования этих библиотек дало возможность значительно повысить функциональность разрабатываемых приложений. Освоил работу с *Microsoft Office Word* программным способом, что является важной задачей для эффективного управления текстовыми документами. Я изучил и применил на практике различные аспекты работы с документами, такие как создание и редактирование текстов, форматирование абзацев, выравнивание и стилизация текста. Более того, я научился создавать и редактировать таблицы, что способствовало лучшему структурированию информации и её визуализации.