**МИНИСТЕРСТВО ТРАНСПОРТА**

**РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ**

**АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ**

**УЧРЕЖДЕНИЕ ВЫСШЕГО**

**ОБРАЗОВАНИЯ**

**“РОССИЙСКИЙ УНИВЕРСИТЕТ**

**ТРАНСПОРТА” (МИИТ)**

**Институт транспортной техники и систем**

**управления**

**(ИТТСУ)**

**Кафедра “Управление и защита информации”**

**Тема: “Лабораторная работа №4”**

**Выполнили студенты группы ТКИ-342**

**Павлючков Даниил Дмитриевич**

**Рудов Лев Андреевич**

**Специальность 10.05.01 “Компьютерная безопасность”**

**Преподаватель: к.т.н. Сафронов А.И.**

**Москва**

**2024**

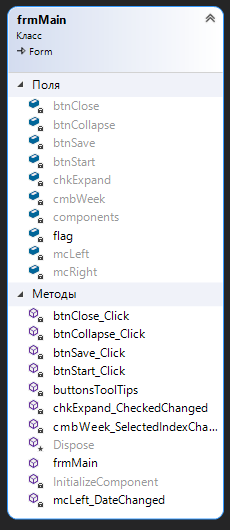
1. Формулировка цели работы:

В интегрированной среде разработки *Microsoft Visual Studio* на языке *Visual C#* разработать программу в режиме *Windows Forms Application*, представляющую собой средство взаимодействия с текстовым файлом, необходимым для отображения на календаре найденных в нём дат (в зависимости от варианта), а также справочной информации.

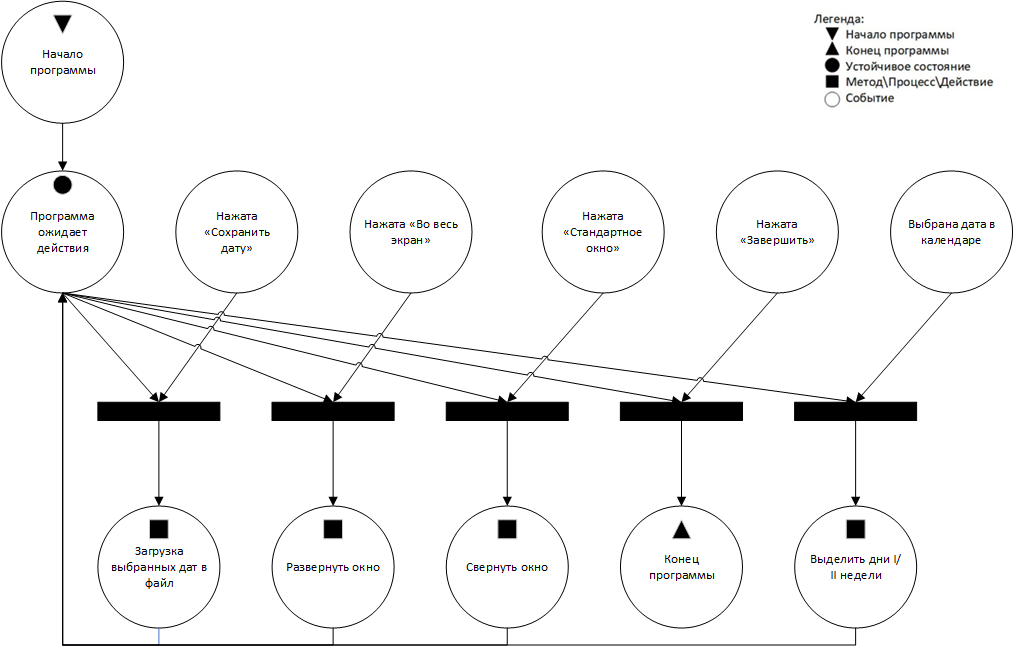
1. Описание задачи:

На левом (или верхнем) календаре отмечается дата. После в комбинированном списке, содержащем строки «I» и «II», указывается номер учебной недели (I / II), в строгом соответствии с действующим расписанием по РУТ (МИИТ), которой принадлежит указанная дата – это входные (исходные) данные. На правом (или нижнем) календаре в ответ выделяются все даты учебного семестра, начиная с выбранной, которые относятся к предстоящим учебным неделям, соответствующим по номерам введённой неделе.

1. Составление диаграммы классов:



1. Составление сети Петри запрограммированного технологического процесса:

  
  
5. Составление схем алгоритмов методов в составе решения, отмеченных на сети Петри в качестве «эффектов» (метка):  
Изображение выглядит как диаграмма, круг, снимок экрана, дизайн

Автоматически созданное описание

6. Листинг (код) программного обеспечения:

using System;

using System.Collections.Generic;

using System.IO;

using System.Windows.Forms;

namespace Lab4

{

public partial class frmMain : Form

{

public frmMain() => InitializeComponent();

bool flag = false;

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

{

(sender as Button).Enabled = false;

mcLeft.Enabled = true;

mcRight.Enabled = true;

}

private void buttonsToolTips(object sender, EventArgs e)

{

ToolTip tip = new ToolTip();

using (StreamReader streamReader = new StreamReader("Свернуть.txt"))

{

tip.SetToolTip(this.btnCollapse, streamReader.ReadToEnd());

}

using (StreamReader streamReader = new StreamReader("Во весь экран.txt"))

{

tip.SetToolTip(this.chkExpand, streamReader.ReadToEnd());

}

using (StreamReader streamReader = new StreamReader("Закрыть.txt"))

{

tip.SetToolTip(this.btnClose, streamReader.ReadToEnd());

}

using (StreamReader streamReader = new StreamReader("Сохранить.txt"))

{

tip.SetToolTip(this.btnSave, streamReader.ReadToEnd());

}

}

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

{

this.Close();

}

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

{

this.WindowState = FormWindowState.Minimized;

}

private void chkExpand\_CheckedChanged(object sender, EventArgs e)

{

(sender as CheckBox).Text = (sender as CheckBox).Checked ?

"Стандартное окно" : "Во весь экран";

if (!flag)

WindowState = FormWindowState.Maximized;

else

WindowState = FormWindowState.Normal;

flag = !flag;

}

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

{

StreamWriter streamWriter = new StreamWriter("Записанные даты.txt", true);

streamWriter.WriteLine(mcLeft.SelectionStart.ToShortDateString());

streamWriter.Close();

}

private void mcLeft\_DateChanged(object sender, DateRangeEventArgs e)

{

DateTime dateStart = DateTime.Parse("02.09.2024");

DateTime dateSelected = mcLeft.SelectionStart.Date;

DateTime dateEnd = DateTime.Parse("01.01.2025");

int week = ((dateSelected - dateStart).Days) / 7;

if (week % 2 == 0 && (dateSelected - dateStart).Days >= 0)

{

cmbWeek.Text = "I";

List<DateTime> boldedDatesList = new List<DateTime>();

for (int i = 0; i < (dateEnd - dateSelected).Days; i++)

{

DateTime tempDate = mcLeft.SelectionStart.Date.AddDays(i);

int tempWeek = (tempDate - dateStart).Days / 7;

if (tempWeek % 2 == 0)

{

boldedDatesList.Add(tempDate);

}

}

DateTime[] boldedDatesArray = boldedDatesList.ToArray();

mcRight.UpdateBoldedDates();

mcRight.AnnuallyBoldedDates = boldedDatesArray;

}

else

{

cmbWeek.Text = "II";

List<DateTime> boldedDatesList = new List<DateTime>();

for (int i = 0; i < (dateEnd - dateSelected).Days; i++)

{

DateTime tempDate = mcLeft.SelectionStart.Date.AddDays(i);

int tempWeek = (tempDate - dateStart).Days / 7;

if (tempWeek % 2 != 0)

{

boldedDatesList.Add(tempDate);

}

}

DateTime[] boldedDatesArray = boldedDatesList.ToArray();

mcRight.UpdateBoldedDates();

mcRight.AnnuallyBoldedDates = boldedDatesArray;

}

}

private void cmbWeek\_SelectedIndexChanged(object sender, EventArgs e)

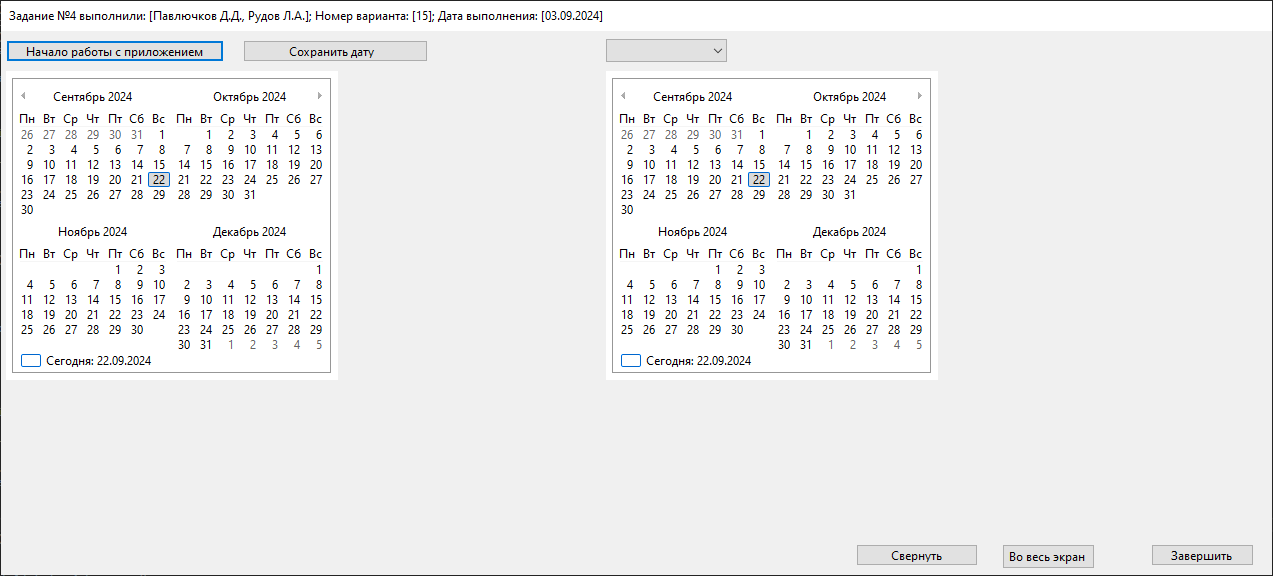
{

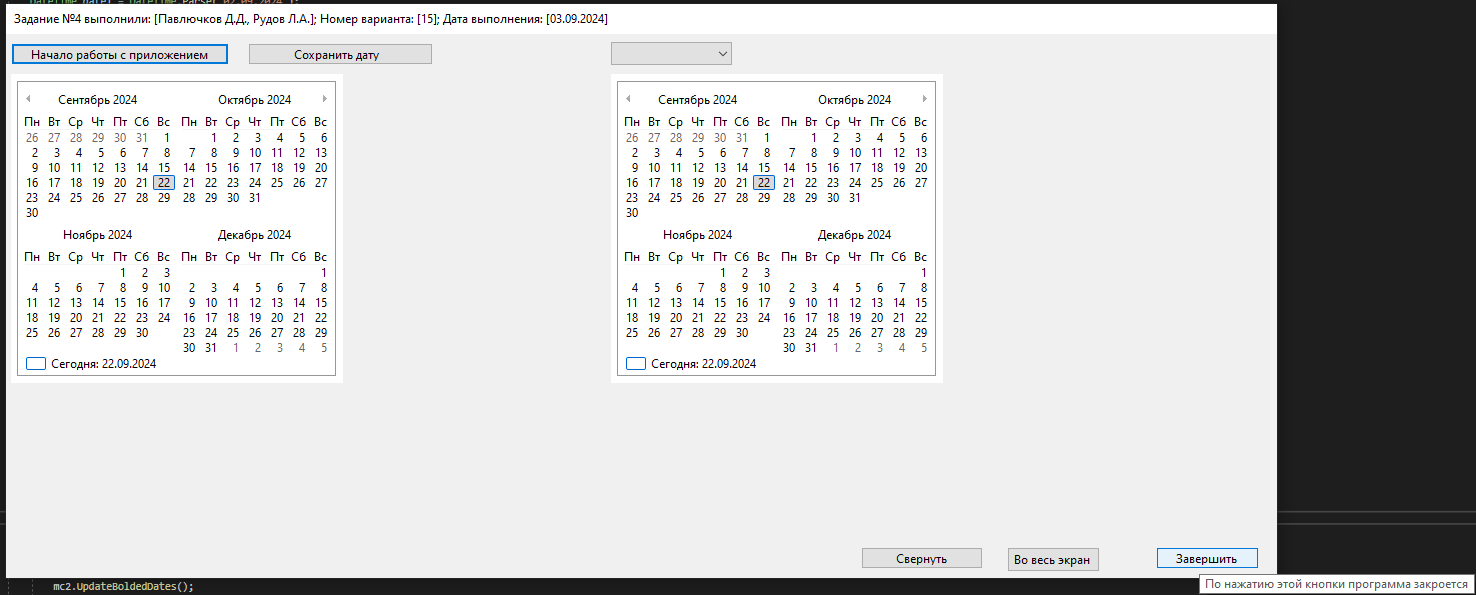
}

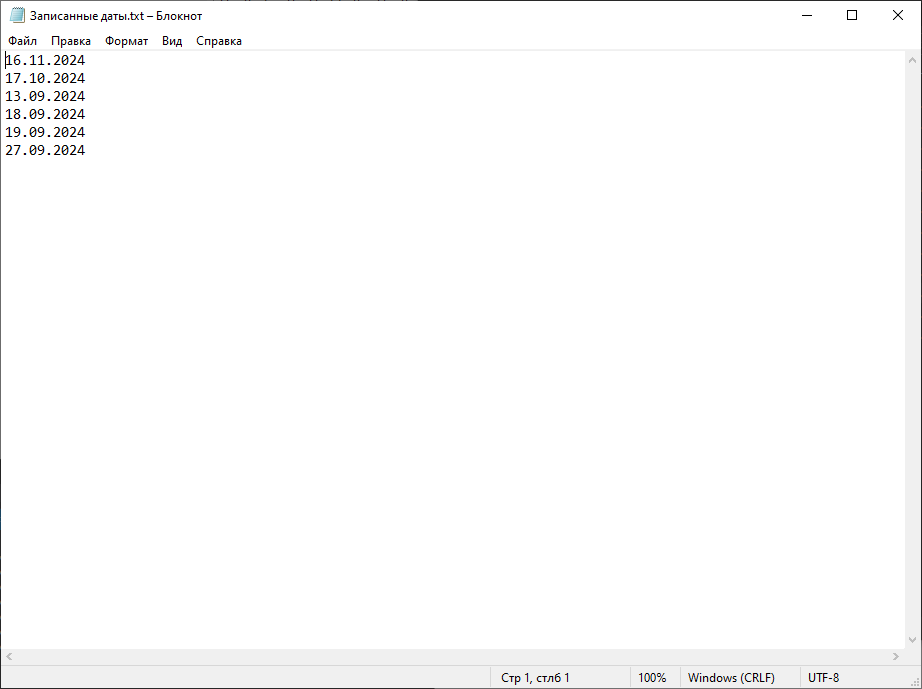
}

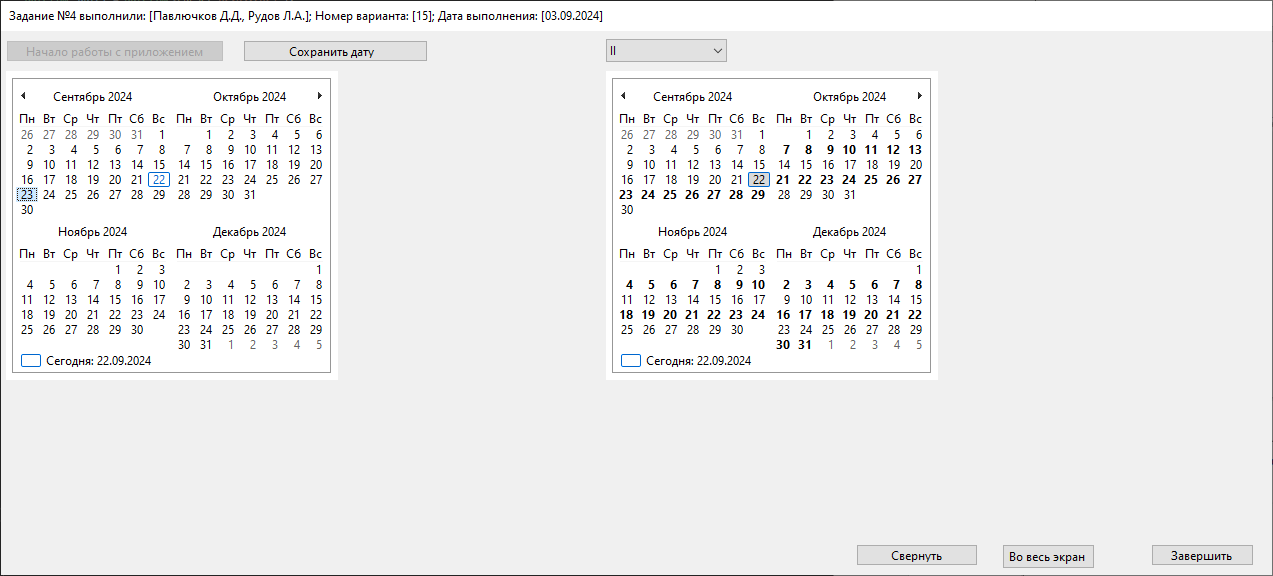
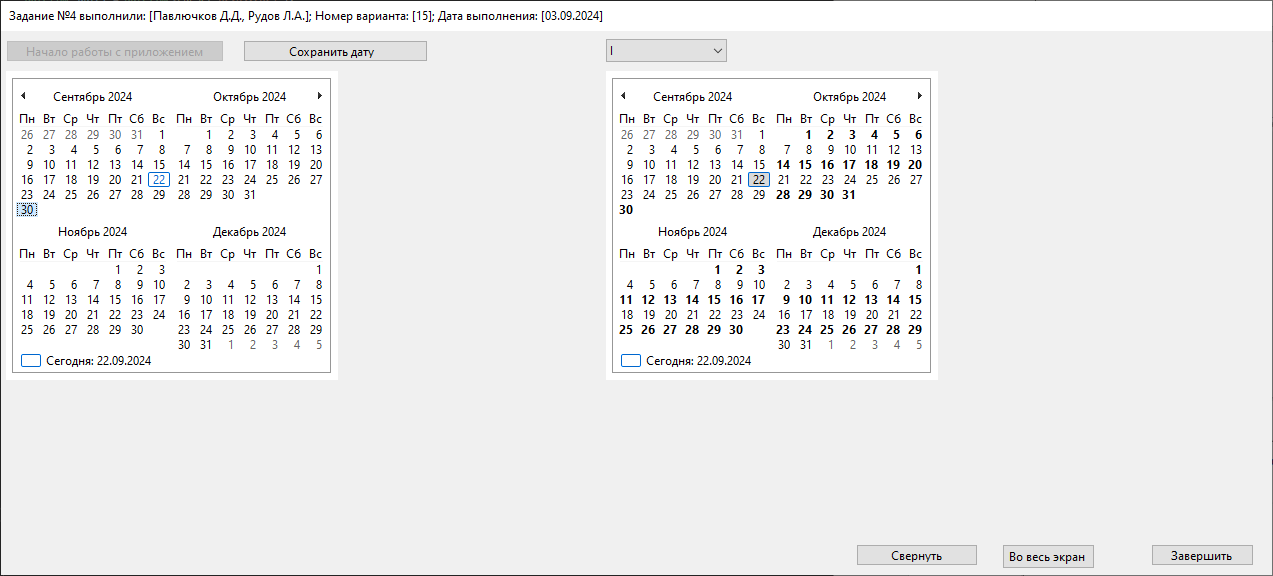
}

7. Графический пользовательский интерфейс:

  
Меню скомпоновано в графической среде

  
У каждого активного элемента есть окно подсказки

  
Файл с записанными датами

Пример выполнения индивидуального задания

8. Вывод:

В ходе выполнения данной работы был получен навык работы с языком программирования с#. Был получен опыт работы по созданию календарей в c# c помощью элемента MonthCalendar, а также с элементом всплывающей подсказки *ToolTip.* Было создано своё графическое приложение, которое выделяет дни I/II недели на календаре, также реализована функция записи дат из календаря в txt файл.