Министерство транспорта Российской Федерации

Федеральное государственное автономное

образовательное учреждение высшего образования

«Российский университет транспорта (МИИТ)»

Институт транспортной техники и систем управления

Кафедра «Управление и защита информации»

Лабораторная работа №4

по дисциплине:

«Методы программирования»

на тему:  
«Работа с календарём»

Выполнил: ст. гр. ТКИ-341

Панаргин В.М.

Вариант №5

Проверил: к.т.н., доцент Сафронов А.И.

Москва – 2024 г.

# **1. Цель работы**

Освоение навыков настройки элемента «Календарь на месяц» (*MonthCalendar*).

# **2. Формулировка задачи**

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.

Предусмотреть в рамках графического пользовательского интерфейса раздел справочной информации, содержащий инструкцию пользователя к разработанному программному обеспечению. Организовать постраничный вывод справочной информации в экранную форму. Выводимые сведения необходимо хранить в текстовых файлах. Предусмотреть запись в файл интересующих дат, сформированных в результате работы программы. Нечётным вариантам необходимо отключить строку состояния с кнопками «*Minimize*», «*Maximize*», «*Close*». Взамен создать на форме авторские кнопки «Свернуть», «Развернуть», «Закрыть». Предусмотреть вызов справки о работе каждого компонента программы путём размещения информации, считанной из файла, в компоненты *ToolTip* (всплывающая подсказка) подсвечиваемых элементов.

На правом (или нижнем) календаре выполнить выделение всех выходных дней указанного календарного года с учётом праздничных дней, принятых постановлением Правительства РФ на текущий календарный год.

# **3. Составление диаграммы классов, входящих в состав решения.**

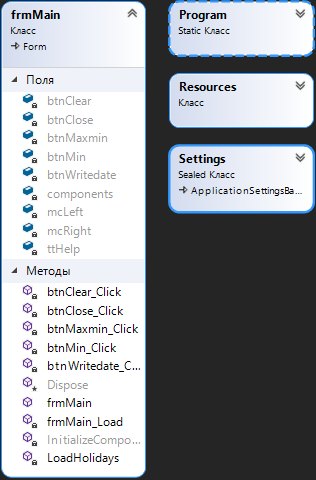


Рисунок 1 – Диаграмма классов

# **4. Составление сети Петри запрограммированного технологического процесса.**

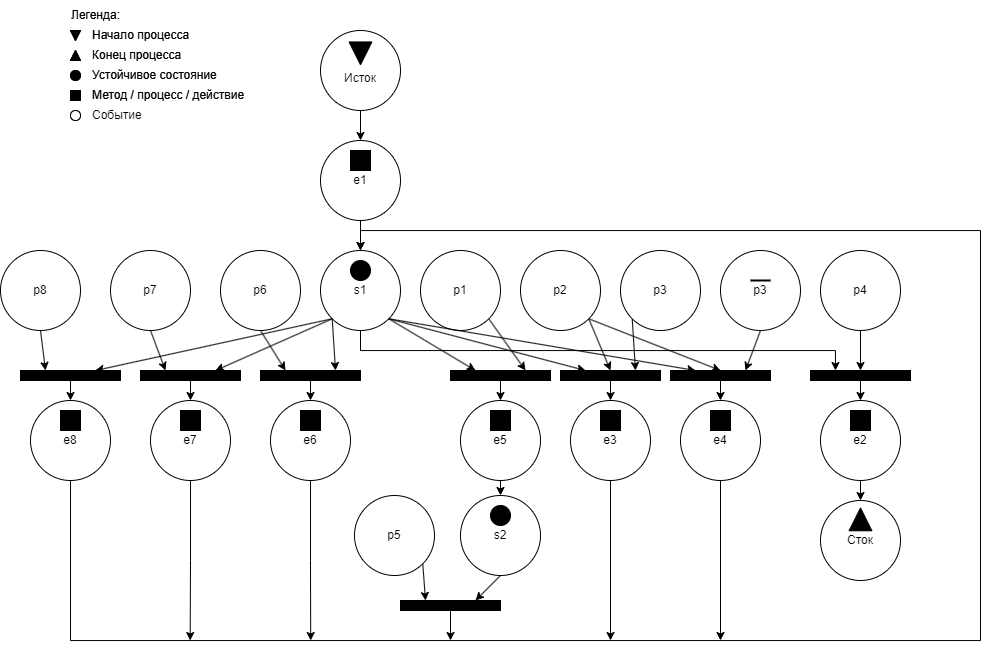


Рисунок 2 - Сеть Петри основной формы

**Описание сети Петри**

- состояния (states)

s1 – форма ожидает действий пользователя

s2 – форма свернута

- действия (effects)

e1 – из файлов загружается необходимая информация;

на правом календаре выделяются выходные дни

e2 – закрытие формы

e3 – форма максимизируется

e4 – форма переходит в нормальное состояние

e5 – форма сворачивается

e6 – запись даты в файл

e7 – очистка файла с датами

e8 – высвечивается информация о выбранном элементе

- события (prompts)

p1 – нажата кнопка «свернуть в трей»

p2 – нажата кнопка «максимизировать/минимизировать»

p3 – форма находится в нормальном состоянии

p4 – нажата кнопка закрытия

p5 – пользователь разворачивает форму

p6 – нажата кнопка записи в файл

p7 – нажата кнопка очистки файла

p8 – пользователь удерживает курсор на одном из элементов

# **5. Составление схем алгоритмов методов в составе решения, отмеченных на сети Петри в качестве «эффектов» (метка ).**



Рисунок 3 – Алгоритм чтения из файлов, выделения выходных дней в правом списке и выбранных дат в левом

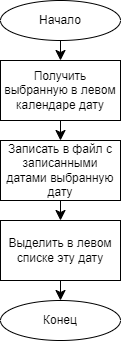


Рисунок 4 - Алгоритм записи информации в файл

# **6. Подбор тестовых примеров.**

Тестовые примеры:

* Проверка кнопки «максимизировать/минимизировать»
* Проверка записи в файл
* Проверка всплывающих уведомлений

**7. Листинг (код) составленного программного обеспечения.**

public partial class frmMain : Form

{

public frmMain()

{

InitializeComponent();

}

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

{

this.WindowState = FormWindowState.Minimized;

}

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

{

if (this.WindowState == FormWindowState.Normal)

{

this.WindowState = FormWindowState.Maximized;

}

else

{

this.WindowState = FormWindowState.Normal;

}

}

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

{

Application.Exit();

}

private void LoadHolidays()

{

StreamReader sr = new StreamReader("E:\\Методы программирования\\Задание 4\\Task4\\holidays.txt");

string[] info = sr.ReadToEnd().Split(';');

this.mcRight.RemoveAllBoldedDates();

foreach (string date in info)

{

this.mcRight.AddBoldedDate(Convert.ToDateTime(date));

}

for (DateTime i = Convert.ToDateTime("01.01.2024"); i <= Convert.ToDateTime("31.12.2024"); i = i.AddDays(1))

{

if (i.DayOfWeek.ToString() == "Saturday" || i.DayOfWeek.ToString() == "Sunday") this.mcRight.AddBoldedDate(i);

}

this.mcRight.UpdateBoldedDates();

sr.Close();

sr = new StreamReader("E:\\Методы программирования\\Задание 4\\Task4\\writtendates.txt");

info = sr.ReadToEnd().Split(new char[] {';'}, StringSplitOptions.RemoveEmptyEntries);

this.mcLeft.RemoveAllBoldedDates();

foreach (string date in info)

{

this.mcLeft.AddBoldedDate(Convert.ToDateTime(date));

}

this.mcLeft.UpdateBoldedDates();

sr.Close();

}

private void frmMain\_Load(object sender, EventArgs e)

{

StreamReader sr = new StreamReader("E:\\Методы программирования\\Задание 4\\Task4\\helpinfo.txt");

string[] info = sr.ReadToEnd().Split(';');

sr.Close();

this.ttHelp.SetToolTip(this.mcLeft, info[0]);

this.ttHelp.SetToolTip(this.mcRight, info[1]);

this.ttHelp.SetToolTip(this.btnClose, info[2]);

this.ttHelp.SetToolTip(this.btnMin, info[3]);

this.ttHelp.SetToolTip(this.btnMaxmin, info[4]);

this.ttHelp.SetToolTip(this.btnWritedate, info[5]);

this.ttHelp.SetToolTip(this.btnClear, info[6]);

this.LoadHolidays();

}

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

{

DateTime selected = this.mcLeft.SelectionStart;

StreamWriter sw = new StreamWriter("E:\\Методы программирования\\Задание 4\\Task4\\writtendates.txt", true);

sw.Write(selected.ToString("d"));

sw.Write(';');

sw.Close();

this.mcLeft.AddBoldedDate(selected);

this.mcLeft.UpdateBoldedDates();

}

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

{

StreamWriter sw = new StreamWriter("E:\\Методы программирования\\Задание 4\\Task4\\writtendates.txt");

sw.Close();

this.mcLeft.RemoveAllBoldedDates();

this.mcLeft.UpdateBoldedDates();

}

}

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

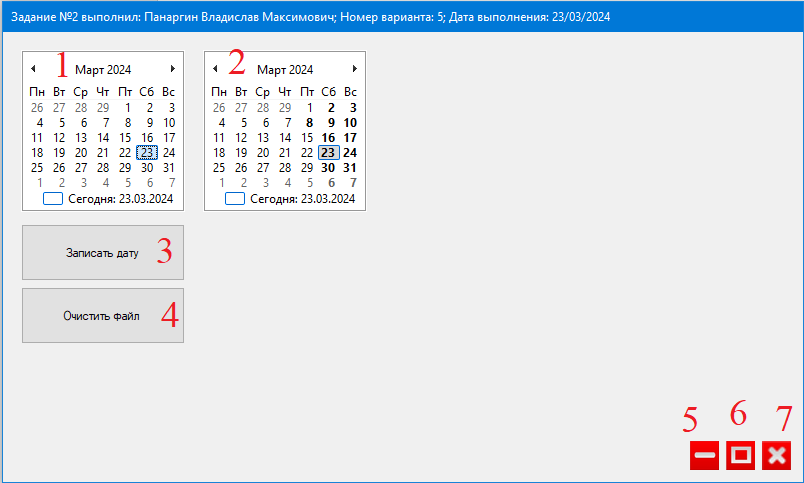


Рисунок 5 - Интерфейс главной формы

1. Левый календарь (для выбора и сохранения дат)
2. Правый календарь (календарь праздников)
3. Кнопка записи выбранной в левом календаре даты в файл
4. Кнопка очистки файла с записанными датами
5. Кнопка «свернуть в трей»
6. Кнопка «максимизировать/минимизировать»
7. Кнопка закрытия

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

1. Заголовок главной формы должен содержать надпись вида: «Задание №4 выполнил: [Фамилия И.О. автора]; Номер варианта: [Номер]; Дата выполнения: [дд/мм/гггг]».



Рисунок 6 - Заголовок главной формы

2. Дата выполнения проставляется в момент, когда программа считается законченной и по ней можно готовить итоговый отчёт.



Рисунок 7 - Дата на момент выполнения задания

3.  Предусмотреть запись в файл интересующих дат, сформированных в результате работы программы (нечётные варианты).

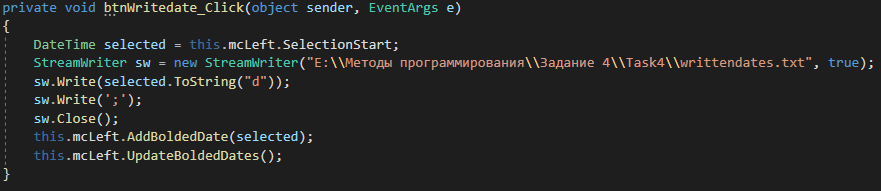


Рисунок 8 - Фрагмент кода с записью даты в файл

4. На форме должны быть размещены два календаря вне зависимости от того, будут ли оба использованы для реализации основного и индивидуального заданий. В случае, когда календарь не используется, необходимо заблокировать его доступность.

См. Рисунок 5

5. Нечётным вариантам необходимо отключить строку состояния с кнопками «*Minimize*», «*Maximize*», «*Close*». Взамен создать на форме авторские кнопки «Свернуть», «Развернуть», «Закрыть».

См. Рисунок 5

6. Предусмотреть вызов справки о работе каждого компонента программы путём размещения информации, считанной из файла, в компоненты *ToolTip* (всплывающая подсказка) подсвечиваемых элементов (нечётные варианты).

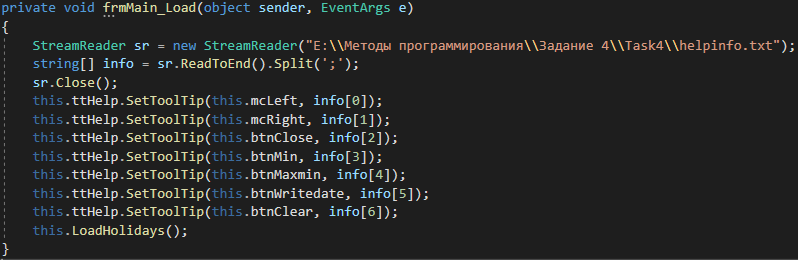


Рисунок 9 - Фрагмент кода с настройкой ToolTip

7. На правом (или нижнем) календаре выполнить выделение всех выходных дней указанного календарного года с учётом праздничных дней, принятых постановлением Правительства РФ на текущий календарный год.

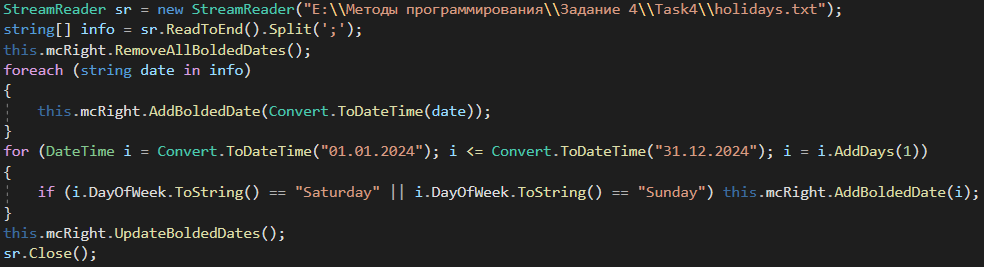


Рисунок 10 - Алгоритм выделения необходимых дат

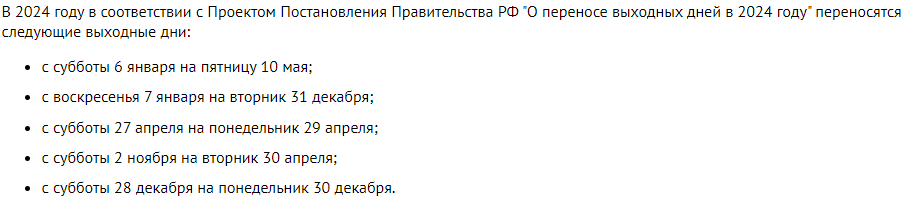


Рисунок 11 - Праздничные дни в согласно постановлению Правительства РФ

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

Тестовые примеры:

* Проверка кнопки «максимизировать/минимизировать»

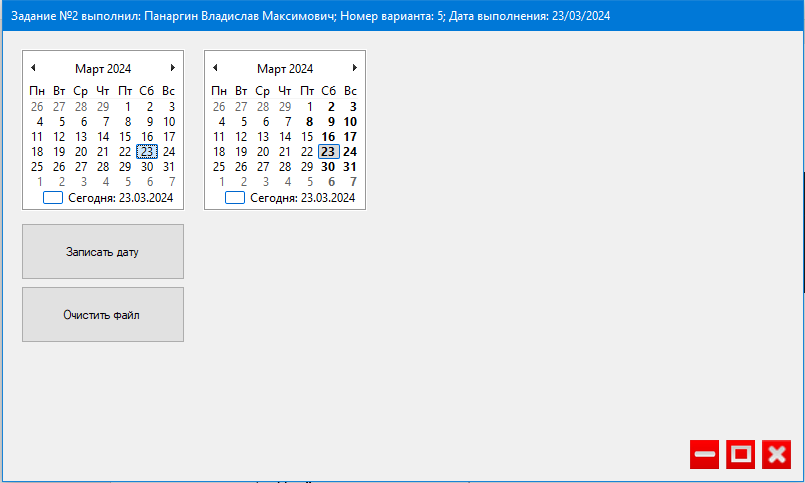


Рисунок 12 - Форма в нормальном состоянии

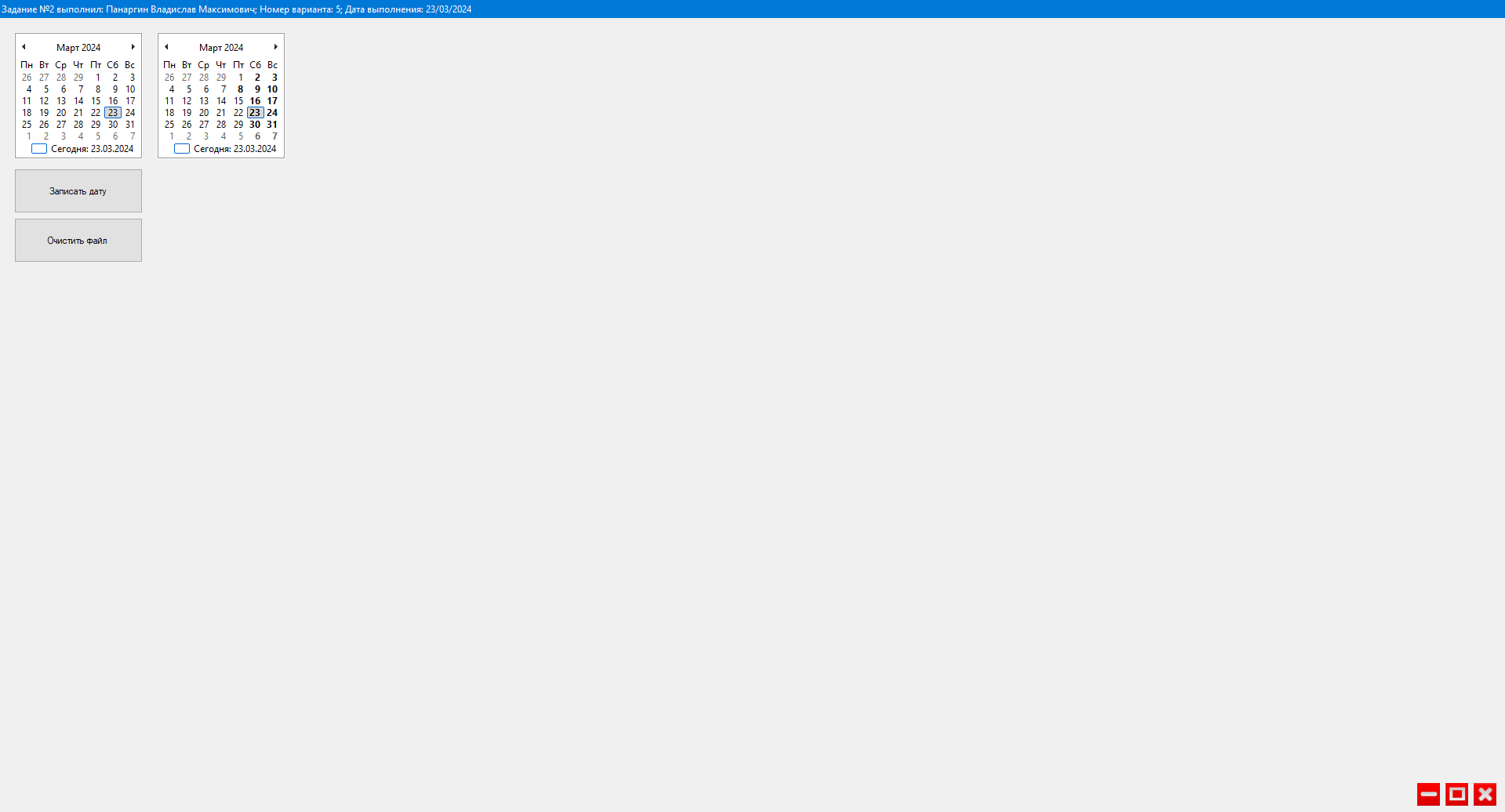


Рисунок 13 - Форма в максимизированном состоянии

* Проверка записи в файл

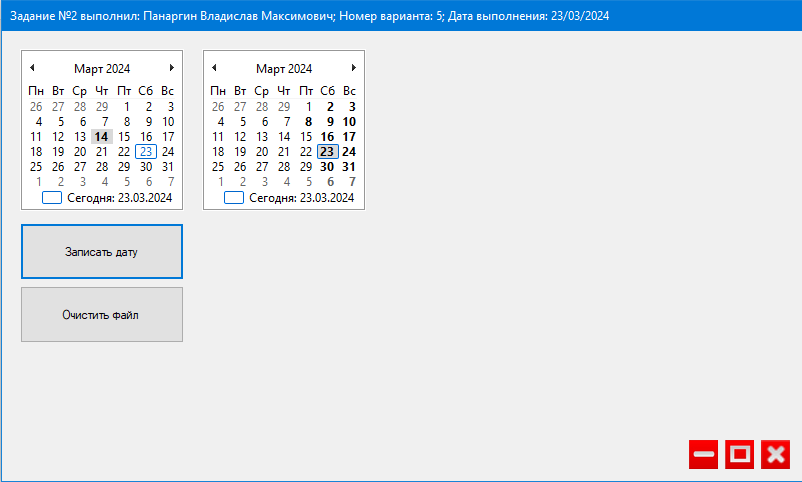


Рисунок 14 - В левом календаре выбрана дата 14 марта 2024 года



Рисунок 15 - В файле сохранилась выбранная дата

* Проверка всплывающих уведомлений

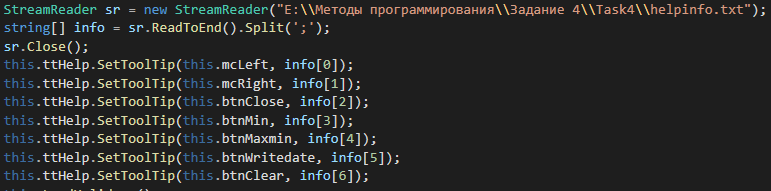


Рисунок 16 - Фрагмент кода с настройкой всплывающих уведомлений

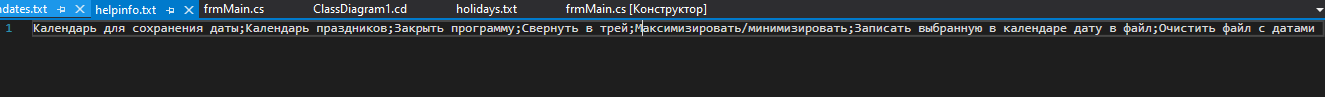


Рисунок 17 - Содержимое файла для настройки всплывающих уведомлений

# **11. Формулировка вывода о проделанной работе**

В рамках данной работы были закреплены навыки разработки визуального пользовательского интерфейса, освоена работа с текстовыми файлами и кодировкой в среде *Microsoft* *Visual Studio*, были получены знания о работе с элементом Календарь на месяц» (*MonthCalendar*).