**Национальный исследовательский университет «Высшая школа экономики»**

**Факультет компьютерных наук**

Департамент

**Программной инженерии**

***Контрольное домашнее задание по дисциплине***

***«Программирование»***

Тема работы: Работа с файлами

Выполнил(а): студент группы 185 (2) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Мануйлов А.А. тел.89122523340

e-mail адрес: aamanuylov@edu.hse.ru

Преподаватель: Чуйкин.Н.К.

Москва, 2019 год. Модуль 3

**Оглавление**

1. **УСЛОВИЕ ЗАДАЧИ 3**
2. **ФУНКЦИИ РАЗРАБАТЫВАЕМОГО ПРИЛОЖЕНИЯ 4**
3. **СТРУКТУРА ПРИЛОЖЕНИЯ 6**
4. **РАСПРЕДЕЛЕНИЕ ИСХОДНОГО КОДА ПО ФАЙЛАМ ПРОЕКТА 9**
5. **КОНТРОЛЬНЫЙ ПРИМЕР И ОПИСАНИЕ РЕЗУЛЬТАТОВ 10**
6. **СООБЩЕНИЯ ПОЛЬЗОВАТЕЛЮ 11**
7. **СПИСОК ЛИТЕРАТУРЫ 14**
8. **ТЕКСТ (КОД) ПРОГРАММЫ 15**

# Условие задачи

Вариант 13

Программа Контрольного домашнего задания представляет собой **WindowsForms** приложение и предназначена для просмотра, обработки и сохранения результатов обработки данных из файла.

**Колледжи 1.1 состояние на 31.10.2013.csv**

1. Требования к основным классам приложения
   1. Основная информация о Колледжах хранится в объектах класса «Колледж». Набор полей класса задаётся полями файла **Колледжи 1.1 состояние на 31.10.2013.csv**, кроме полей, содержащих информацию об административном округе и районе, представленых полем типа **Район**. Класс **Колледж** находится в *отношении агрегации* с классом **Район**. Один из методов класса возвращает признак **true**, если колледж государственный и **false** – в противном случае.
   2. Класс **Район** представляет информацию об административном расположении колледжа, заданном полями CSV-файла: **okrug**, **rayon**.
   3. Дополнительные классы, необходимые для решения задачи.
2. Приложение должно поддерживать следующие функции
   1. Открыть CSV-файл (**\*.csv**) с исходными данными и проверить корректность данных в нём.
   2. Загрузить данные из CSV-файла в объекты классов **Колледж**, **Район** (если объект **Район** с данными об определённом районе существует, то он является общим для всех объектов **Колледж** этого района) и др.
   3. Отобразить данные из объектов в оконной форме.
   4. Создать новую запись о Колледже.
   5. Удалить уже существующую запись о Колледже.
   6. Отредактировать существующую запись о Колледже.
   7. Отсортировать данные по полям: **name**, **vid\_uchrezhdenia**
   8. Отфильтровать данные по признаку государственности/не государственности для района, в котором расположен колледж, выбранный пользователем из списка.
   9. Отфильтровать данные по полям: **okrug, rayon**
   10. Сохранять результаты редактирований, сортировок и фильтраций в CSVфайл. *Режимы* *сохранения в файл*: создание нового файла, замена содержимого уже существующего файла, добавление сохраняемых данных к содержимому существующего файла.
3. Требования к интерфейсу
   1. При управлении файлом (загрузка, сохранение) использовать **OpenFileDialog** и **SaveFileDialog**.
   2. Для отображения данных использовать сетку **DataGridView.** Количество отображаемых в сетке элементов (***N***) выбирается пользователем, ***N* > 1** и не превышает количества записей в файле **Колледжи 1.1 состояние на 31.10.2013.csv**

4. Требования к устойчивости приложения

4.1. В случае ошибок открывания/сохранения файла или некорректных данных программа должна выводить сообщение.

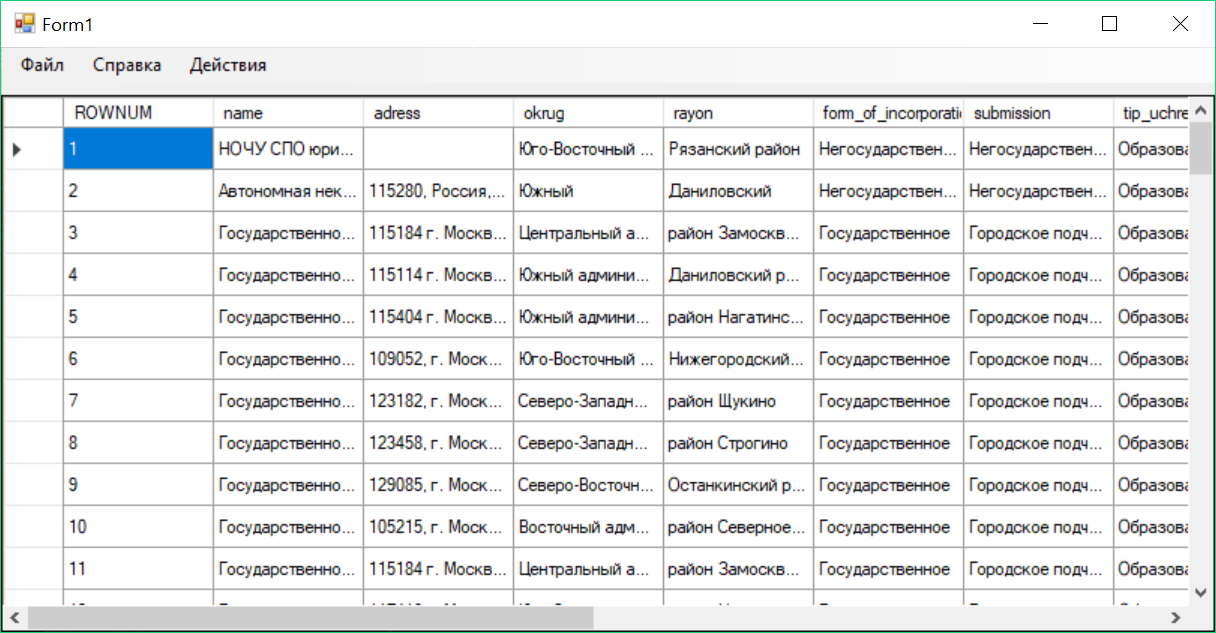
4.2. Аварийные ситуации должны обрабатываться, пользователю должны выводиться информативные сообщения.

# Функции разрабатываемого приложения

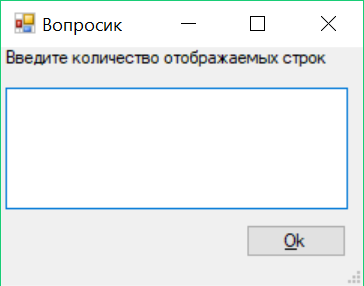
* 1. Варианты использования. Данное приложение может использоваться в административном учете, структурировании и корректировании информации о колледжах Москвы.

2.2. Описание интерфейса пользователя: Пользователь может открыть csv файл с информацией о колледжах, удалять записи, добавлять новые или редактировать существующие, также пользователь может совершать предусмотренные ТЗ фильтрации и сортировки и сохранять документ.

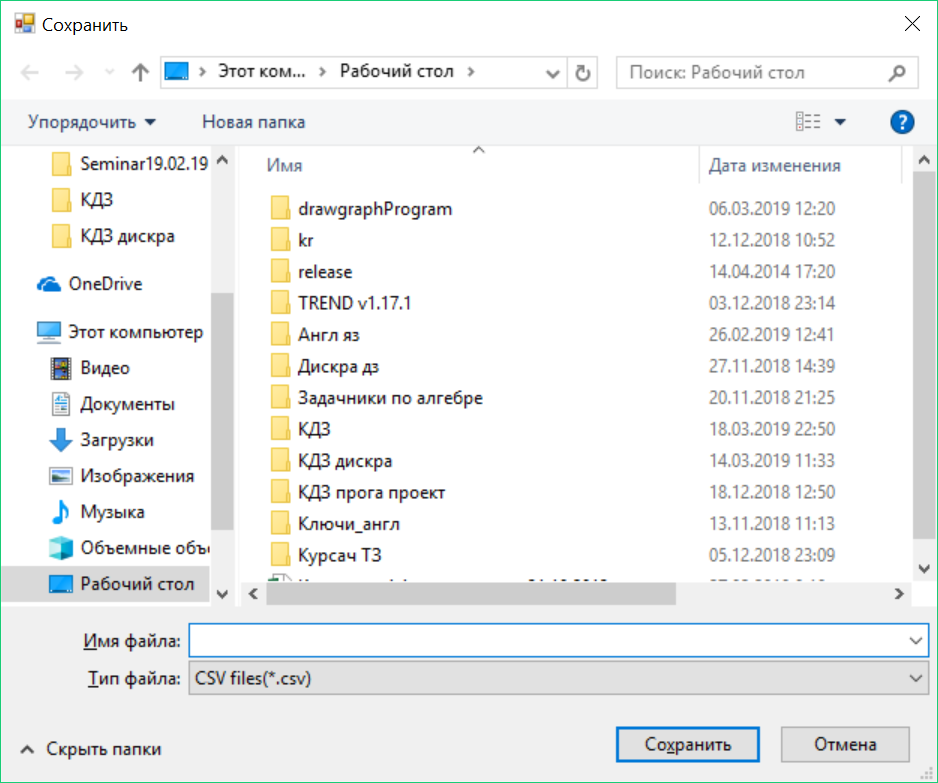
2.3. Основное окно программы



2.5. Окно для ввода количества отображаемых строк

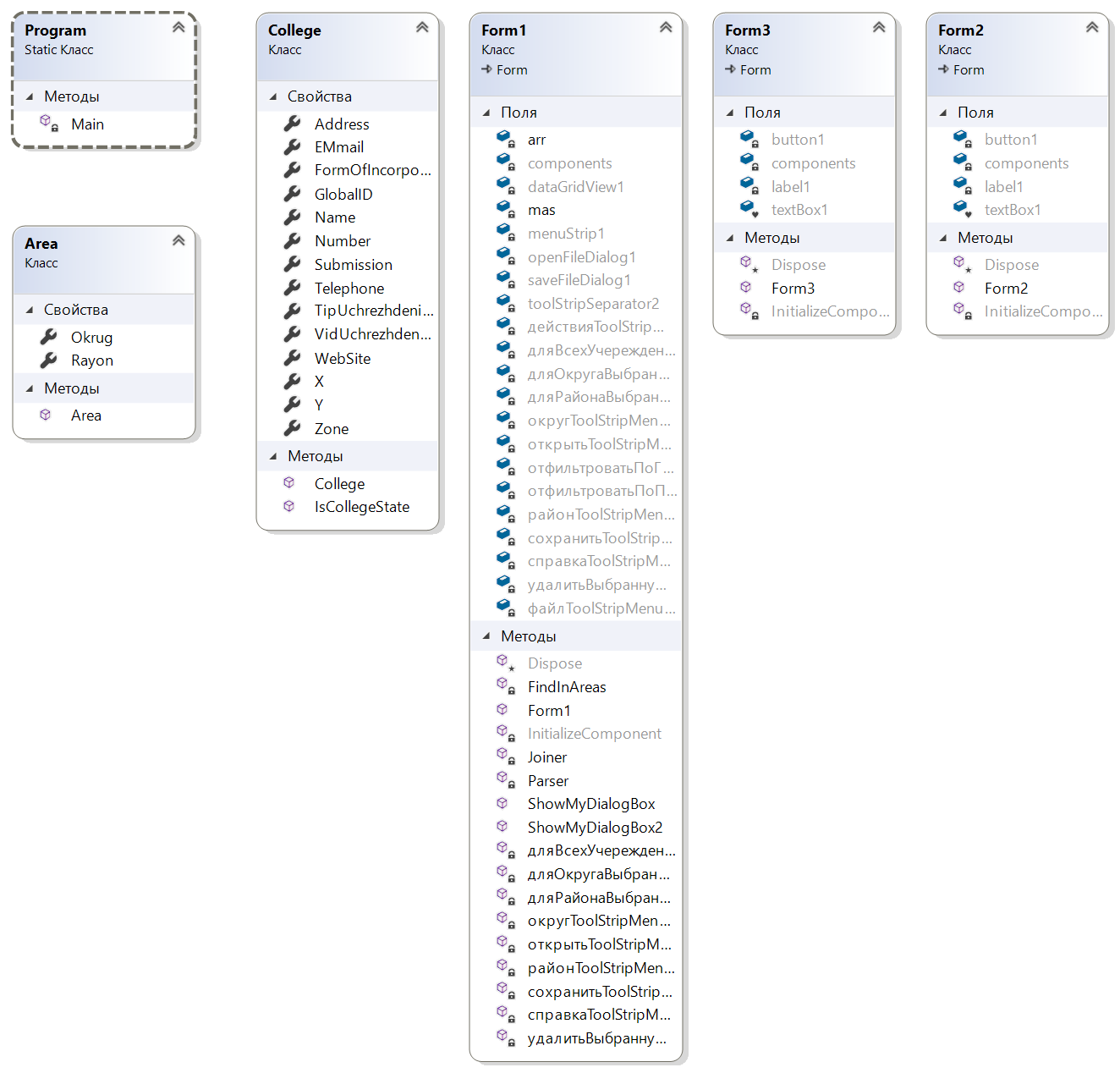


2.6. Окно для открытия .csv файла



# Структура приложения

* 1. **Диаграмма классов:**



**3.2. Описание классов, их полей и методов Классы:**

* **class Program** класс, содержащий метод Main()
* **public partial class Form1: Form** форма, содержащая основные методы для занесения информации в таблицу.
* **public partial class From2: Form** класс с окном ввода количества выводимых строк.
* **public partial class From3: Form** класс с окном ввода подстроки для фильтрации района и округа.
* **Public class College** класс содержащий поля, соответствующие столбцам таблицы.
* **Public class Area** класс состоящий в состоянии агрегации с классом College содержащий поля район и округ.

**Методы:**

* **public partial class Form1: Form** 
  + **private int FindIsAreas** метод, возвращающий индекс экземпляра Area, если он есть в массиве arr и -1 иначе.
  + **private void открытьToolStripMenuItem\_Click** метод открывающий файл, читающий из него информацию и заносящий ее в таблицу
  + **private string[] Parser** разделяет строку на массив по колонкам
  + **private void удалитьВыбраннуюСтрокуToolStripMenuItem\_Click** метод, удаляющий выбранную строку в таблице
  + **public string ShowMyDialogBox** метод показывает окно для ввода количества отображаемых строк
  + **private void справкаToolStripMenuItem\_Click** выводит на экран информацию о программе
  + **private void сохранитьToolStripMenuItem\_Click** метод для сохранения информации в таблице в csv файл
  + **private string Joiner** метод создающий строку для сохранения в файл
  + **private void дляВсехУчережденийToolStripMenuItem\_Click** метод для фильтрации гос-ти/не гос-ти для всех значений
  + **private void дляРайонаВыбранногоКолледжаToolStripMenuItem\_Click** метод для фильтрации гос-ти/не гос-ти для района выбранного колледжа
  + **private void дляОкругаВыбранногоКолледжаToolStripMenuItem\_Click** метод для фильтрации гос-ти/не гос-ти для округа выбранного колледжа
  + **public string ShowMyDialogBox2** метод показывает окно для ввода подстроки фильтрации
  + **private void районToolStripMenuItem\_Click** метод для фильтрации по району
  + **private void округToolStripMenuItem\_Click** метод для фильтрации по округу
* **internal class College**

o **public bool IsCollegeState** метод определяющий является ли колледж государственным

**Поля:**

* **partial class Form1 : Form**
  + **private List<College> mas** для хранения всех колледжей
  + **private List<Area> arr** для хранения всех районов
* **public class College**
  + **public Area zone**
  + **public string number**
  + **public string name**
  + **public string address**
  + **public string formOfIncorporation**
  + **public string submission**
  + **public string tipUchrezhdeniya**
  + **public string vidUchrezhdeniya**
  + **public string telephone**
  + **public string webSite**
  + **public string eMail**
  + **public string x**
  + **public string y**
  + **public string globalID**
* **public class Area**
  + **public string okrug**
  + **public string rayon**

# Распределение исходного кода по файлам проекта

1. Form1.cs – основное окно программы
2. Form2.cs – окно для ввода количества отображаемых строк
3. Form3.cs – окно для ввода подстроки для фильтрации
4. College.cs – класс содержащий поля, соответствующие столбцам .csv файла
5. Area.cs - район и округ
6. Program.cs - начало программы

# Контрольный пример и описание результатов

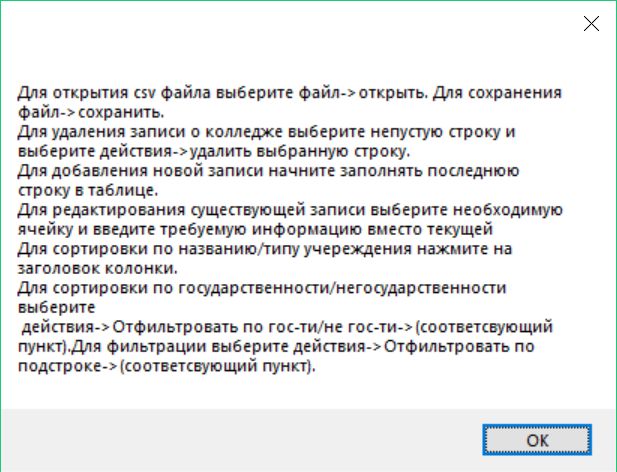
Программа Контрольного домашнего задания представляет собой WindowsForms приложение и предназначена для просмотра, обработки и сохранения результатов обработки данных из файла.

Результаты тестирования:

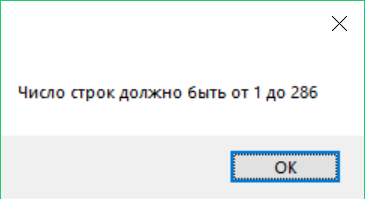
1. При запуске программы появляется пустое окно.
2. Для открытия файла нужно выбрать файл->открыть.
3. Можно произвести действия с файлом (сортировка фильтрация, удаление строк).
4. Можно редактировать таблицу.
5. После необходимых преобразований можно сохранить таблицу в csv файл.

# Сообщения пользователю

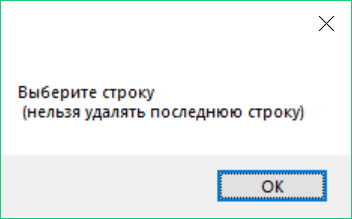
* 1. При нажатии на справку:



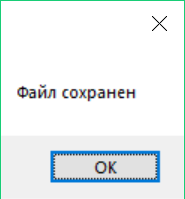
* 1. При некорректном вводе числа отображаемых строк:



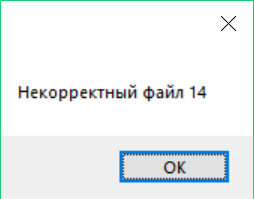
* 1. При попытке удаления последней строки:



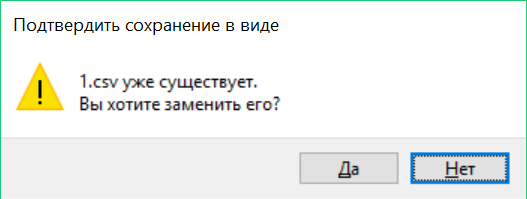
* 1. При сохранении файла:



* 1. При попытке открыть некорректный файл:



* 1. При попытке заменить файл при сохранении:



# Список литературы

1. C# 4.0. Полное руководство

Герберт Шилдт 2011

1. Руководство по C# [Электронный ресурс]

URL: https://msdn.microsoft.com/ru-ru/ (14.03.19)

1. Полное руководство по C# 7.0. [Электронный ресурс]

URL: <https://metanit.com/sharp/> (15.03.19)