МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное АВТОНОМНОЕ образовательное учреждение высшего образования

«МУРМАНСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

**Кафедра математики, информационных систем и программного обеспечения**

**КУРСОВАЯ РАБОТА № 2**

по дисциплине «Технологии программирования»

**«Деканат»**

|  |  |  |  |
| --- | --- | --- | --- |
| Выполнил студент |  | | |
|  | (фамилия, И.О.) | | |
| Группа |  | | |
|  | (номер группы) | | |
| Дата сдачи пояснительной записки | | |  |
| Подпись студента | |  | |
| Преподаватель | |  | |
|  | | (фамилия, И.О.) | |
| Дата защиты | |  | |
| Оценка | |  | |
| Подпись преподавателя | |  | |

Мурманск 2021

Оглавление

[**Введение** 3](#_Toc88399372)

[**Описание предметной области** 4](#_Toc88399374)

[**Функциональные требования** 5](#_Toc88399377)

[**Описание программных решений** 6](#_Toc88399389)

[**Структура программного приложения** 13](#_Toc88399390)

[**Описание пользовательского интерфейса** 14](#_Toc88399392)

[**1.** **Основная форма** 14](#_Toc88399393)

[**2.** **Претенденты на повышенную стипендию** 15](#_Toc88399396)

[**3.** **Выбор сессии для построения круговой диаграммы** 16](#_Toc88399399)

[**4.** **График изменений среднего балла студента** 16](#_Toc88399402)

[**5.** **Круговая диаграмма с оценками по Информатике** 17](#_Toc88399405)

[**6.** **Диаграмма количества студентов по всем специальностям** 18](#_Toc88399408)

[**Описание тестовых сценариев** 19](#_Toc88399411)

[**Список использованных источников** 23](#_Toc88399460)

**Введение**

Целью курсовой работы является разработка системы учета и анализа результатов сессий для деканата. В качестве среды для разработки была выбрана Microsoft Visual Studio. В качестве целевой рабочей среды был выбран .NET Framework версии 4.7.2. В ходе разработки были получены практические навыки разработки на языке C# с помощью технологий объектно-ориентированного программирования.

**Описание предметной области**

Предметной областью курсовой работы является анализ результатов сессий студентов за все время обучения. На основании результатов сессий можно выявить студентов, которые могут претендовать на повышенную стипендию, или наоборот имеют слишком много задолженностей. Также, можно использовать полученную информацию для реализации графиков, которые будут демонстрировать различную статистику успеваемости как определенных студентов, так и всех вместе взятых.

Для полноценного анализа нужно разработать систему способную хранить, редактировать и использовать данные о результатах сессий.

**Функциональные требования**

Подсистема должна решать следующие задачи:

1. Создание файла.
2. Дополнение файла.
3. Корректировка данных.
4. Формирование ответов на запросы пользователя:
   1. Выводить результаты всех сессий по заданной фамилии студента.
   2. Вывести фамилии студентов, имеющих по результатам текущей сессии три и более хвостов.
   3. Вывести в алфавитном порядке ФИО и номера зачетных книжек студентов, претендующих по результатам последней сессии на повышенную стипендию.
   4. Строить график изменения среднего балла студента за все сессии, задавая студента по номеру зачетной книжки.
   5. Строить круговую диаграмму для выбранной сессии, отражающую соотношение оценок на экзамене по Информатике.
   6. Строить столбиковую диаграмму распределения числа студентов по специальностям по курсам за последний учебный год.

**Описание программных решений**

Таблица 1

Описание формы «MainForm»

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Родитель** | Form | **Интерфейсы** | | - |
| **Назначение** | Главная форма для работы с системой | | | |
| *Поля* | | | | |
| **Модификатор доступа** | **Тип** | **Наименование** | | |
| Private | OpenFileDialog | openFileDialog1 | | |
| Private | SaveFileDialog | saveFileDialog1 | | |
| Private | DataGridView | dataGridView1 | | |
| Private | DataGridView | dataGridView | | |
| Private | Panel | panel1 | | |
| Private | Panel | panel2 | | |
| Private | Panel | panel3 | | |
| Private | Label | menuLabel | | |
| Private | Label | labelId | | |
| Private | FileSystemWatcher | fileSystemWatcher1 | | |
| Private | Button | closeButton | | |
| Private | Button | newDocument | | |
| Private | Button | newLine | | |
| Private | Button | backButton | | |
| Private | Button | buttonLoad | | |
| Private | Button | buttonSave | | |
| Private | Button | debtors | | |
| Private | Button | buttonUpperStep | | |
| Private | Button | buttonDiagram | | |
| Private | Button | buttonGraph | | |
| Private | Button | buttonDiagrams | | |
| Private | Button | buttonDelete | | |
| Private | DataGridViewTextBoxColumn | id | | |
| Private | DataGridViewTextBoxColumn | lastName | | |
| Private | DataGridViewTextBoxColumn | name | | |
| Private | DataGridViewTextBoxColumn | surName | | |
| Private | DataGridViewTextBoxColumn | recordBookNumber | | |
| Private | DataGridViewTextBoxColumn | recordNumber | | |
| Private | DataGridViewButtonColumn | performance | | |
| Private | DataGridViewComboBoxColumn | specialization | | |
| Private | DataGridViewComboBoxColumn | subject | | |
| Private | DataGridViewComboBoxColumn | session | | |
| Private | DataGridViewComboBoxColumn | year | | |
| Private | DataGridViewComboBoxColumn | value | | |
| Private | IContainer | components | | |
| *Методы* | | | | |
| **Модификатор доступа** | **Возвращаемый тип** | **Наименование** | **Список параметров (тип параметра - наименование)** | |
| Private | Void | closeButton\_Click | (object – sender, EventArgs – e) | |
| Private | Void | newDocument\_Click | (object – sender, EventArgs – e) | |
| Private | Void | newLine\_Click | (object – sender, EventArgs – e) | |
| Private | Void | dataGridView1\_CellClick | (object – sender, DataGridViewCellEventArgs – e) | |
| Private | Void | backButton\_Click | (object – sender, EventArgs – e) | |
| Private | Void | swapGrids | - | |
| Private | Void | hideRows | - | |
| Private | Void | buttonSave\_Click | (object – sender, EventArgs – e) | |
| Private | DataGridViewButtonColumn | buttonCreate | - | |
| Private | Void | buttonLoad\_Click | (object – sender, EventArgs – e) | |
| Private | Void | debtors\_Click | (object – sender, EventArgs – e) | |
| Private | Void | buttonUpperStep\_Click | (object – sender, EventArgs – e) | |
| Private | Void | buttonGraph\_Click | (object – sender, EventArgs – e) | |
| Private | Void | buttonDiagram\_Click | (object – sender, EventArgs – e) | |
| Private | Void | buttonDiagrams\_Click | (object – sender, EventArgs – e) | |
| Private | Void | buttonDelete\_Click | (object – sender, EventArgs – e) | |
| Public | DataGridViewComboBoxColumn | getSpecialicaztion | - | |
| Public | DataGridViewComboBoxColumn | getSession | - | |
| Public | DataGridViewComboBoxColumn | getValue | - | |
| Public | DataGridViewComboBoxColumn | getYear | - | |
| Public | DataGridViewComboBoxColumn | getSubject | - | |
| Public | DataGridView | getStudentsTable | - | |
| Public | DataGridView | getValuesTable | - | |
| Private | Void | Dispose | (bool – disposing) | |
| Private | Void | InitializeComponent | - | |

Таблица 2

Описание формы «BestStudentsTable»

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Родитель** | Form | **Интерфейсы** | | - |
| **Назначение** | Таблица с ФИО и зачеткой студентов, претендующих на повышенную стипендию | | | |
| *Поля* | | | | |
| **Модификатор доступа** | **Тип** | **Наименование** | | |
| Private | DataGridView | dataGridView1 | | |
| Private | DataGridViewTextBoxColumn | lastName | | |
| Private | DataGridViewTextBoxColumn | name | | |
| Private | DataGridViewTextBoxColumn | surName | | |
| Private | DataGridViewTextBoxColumn | id | | |
| Private | Panel | panel3 | | |
| Private | Label | menuLabel | | |
| Private | Button | closeButton | | |
| Private | IContainer | components | | |
| *Методы* | | | | |
| **Модификатор доступа** | **Возвращаемый тип** | **Наименование** | **Список параметров (тип параметра - наименование)** | |
| Public | Void | gridFiller | (DataGridView – dataGrid) | |
| Private | Void | closeButton\_Click | (object – sender, EventArgs – e) | |
| Private | Void | Dispose | (bool – disposing) | |
| Private | Void | InitializeComponent | - | |

Таблица 3

Описание формы «GraphPainter»

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Родитель** | Form | **Интерфейсы** | | - |
| **Назначение** | Форма для выбора сессии или студента по номеру зачетки, для вызова другой формы | | | |
| *Поля* | | | | |
| **Модификатор доступа** | **Тип** | **Наименование** | | |
| Private | Panel | panel3 | | |
| Private | Panel | panel2 | | |
| Private | Label | menuLabel | | |
| Private | Button | closeButton | | |
| Private | Button | buttonOK | | |
| Private | Button | buttonCancel | | |
| Private | TextBox | textBoxStudent | | |
| Private | ComboBox | comboBoxYear | | |
| Private | ComboBox | comboBoxSession | | |
| Private | IContainer | components | | |
| Private | DataGridView | dgv1 | | |
| Private | DataGridView | dgv2 | | |
| *Методы* | | | | |
| **Модификатор доступа** | **Возвращаемый тип** | **Наименование** | **Список параметров (тип параметра - наименование)** | |
| Public | Void | setChart | (int[,] – data, string[] – countries, int – x, imt - y) | |
| Public | Void | graphPrinter | (DataGridView – dgv1, DataGridView – dgv2) | |
| Public | Void | informatic | (DataGridViewComboBoxColumn – session, DataGridViewComboBoxColumn – year, DataGridView – dgv2) | |
| Private | Void | buttonOK\_Click | (object – sender, EventArgs – e) | |
| Private | Void | Dispose | (bool – disposing) | |
| Private | Void | InitializeComponent | - | |

Таблица 4

Описание формы «InformaticDiagram»

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Родитель** | Form | **Интерфейсы** | | - |
| **Назначение** | Круговая диаграмма для выбранной сессии с информацией о полученных оценках за экзамен по Информатике | | | |
| *Поля* | | | | |
| **Модификатор доступа** | **Тип** | **Наименование** | | |
| Private | IContainer | components | | |
| Private | Chart | chart1 | | |
| *Методы* | | | | |
| **Модификатор доступа** | **Возвращаемый тип** | **Наименование** | **Список параметров (тип параметра - наименование)** | |
| Public | Void | chartSetter | (int[] – data, string – session) | |
| Private | Void | Dispose | (bool – disposing) | |
| Private | Void | InitializeComponent | - | |

Таблица 5

Описание формы «SpecializationDiagrams»

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Родитель** | Form | **Интерфейсы** | | - |
| **Назначение** | Диаграммы с кол-вом студентов по всем специальностям | | | |
| *Поля* | | | | |
| **Модификатор доступа** | **Тип** | **Наименование** | | |
| Private | IContainer | components | | |
| Private | Chart | chart1 | | |
| *Методы* | | | | |
| **Модификатор доступа** | **Возвращаемый тип** | **Наименование** | **Список параметров (тип параметра - наименование)** | |
| Public | Void | diagrammSetter | (string[] – allSpecializations, int[] – specCounters) | |
| Private | Void | Dispose | (bool – disposing) | |
| Private | Void | InitializeComponent | - | |

Таблица 6

Описание класса «StudentGraph»

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Родитель** | Form | **Интерфейсы** | | - |
| **Назначение** | График изменения среднего балла студента за все сессии | | | |
| *Поля* | | | | |
| **Модификатор доступа** | **Тип** | **Наименование** | | |
| Private | IContainer | components | | |
| Private | Chart | chart1 | | |
| *Методы* | | | | |
| **Модификатор доступа** | **Возвращаемый тип** | **Наименование** | **Список параметров (тип параметра - наименование)** | |
| Public | Void | diagrammSetter | (string[] –allSpecializations, int[] – specCounters) | |
| Private | Void | Dispose | (bool – disposing) | |
| Private | Void | InitializeComponent | - | |

Таблица 7

Описание класса «TableFiller»

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Родитель** | Object | **Интерфейсы** | | - |
| **Назначение** | Класс с методами, которые реализуют заполнение таблиц со студентами и с результатами сессий | | | |
| *Поля* | | | | |
| **Модификатор доступа** | **Тип** | **Наименование** | | |
| Private | MainForm | mainForm | | |
| *Методы* | | | | |
| **Модификатор доступа** | **Возвращаемый тип** | **Наименование** | **Список параметров (тип параметра - наименование)** | |
| Public | Void | tablesFiller | (string[] – info) | |
| Public | Void | tableSaveFileFiller | (StreamWriter – streamWriter) | |

Таблица 8

Описание класса «TableLoader»

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Родитель** | Object | **Интерфейсы** | | - |
| **Назначение** | Класс с методами, которые реализуют загрузка данных в файл и выгрузку данных из файла | | | |
| *Поля* | | | | |
| **Модификатор доступа** | **Тип** | **Наименование** | | |
| Private | MainForm | mainForm | | |
| Private | StreamWriter | streamWriter | | |
| Private | StreamReader | streamReader | | |
| Private | TableFiller | tableFiller | | |
| Private | Stream | stream | | |
| *Методы* | | | | |
| **Модификатор доступа** | **Возвращаемый тип** | **Наименование** | **Список параметров (тип параметра - наименование)** | |
| Public | Void | loadTable | - | |
| Public | Void | saveTable | - | |

**Структура программного приложения**

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 1 - Обозреватель решений

**Описание пользовательского интерфейса**

1. **Основная форма**

В основной форме можно создавать, редактировать и удалять данные, также сохранять и загружать файлы, которые в себе хранят все нужные данные. Также на основной форме расположены кнопки для вывода графиков, диаграмм и таблиц на основе обрабатываемых в программе данных.

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 2 - Основная форма

1. **Претенденты на повышенную стипендию**

Форма “Претенденты на повышенную стипендию” реализует таблицу с информацией о студентах претендующих, по результатам последней сессии, на повышенную стипендию.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 3 – Претенденты на повышенную стипендию

1. **Выбор сессии для построения круговой диаграммы**

Форма “Выбор сессии для построения круговой диаграммы” представляет собой форму, в которой можно выбрать год и сезон сессии для построения круговой диаграммы.

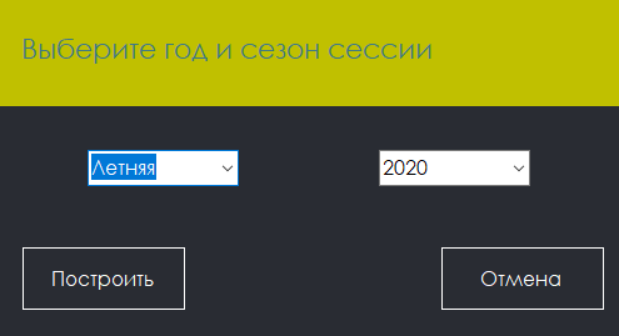


Рисунок 4 – Выбор сессии

1. **График изменений среднего балла студента**

Форма “График изменений среднего балла студента” представляет собой график изменений среднего балла студента по всем сессиям.

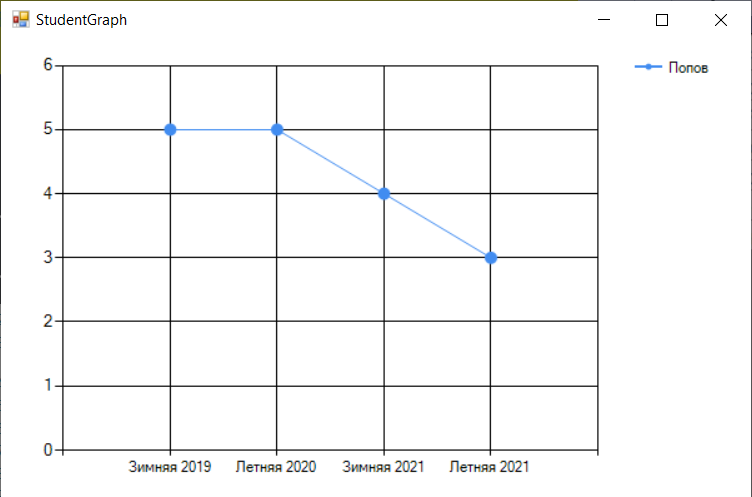


Рисунок 5 - Графики изменения среднего балла

1. **Круговая диаграмма с оценками по Информатике**

Форма “Круговая диаграмма с оценками по Информатике” представляет собой круговую диаграмму с информацией об оценках по Информатике в выбранной сессии.

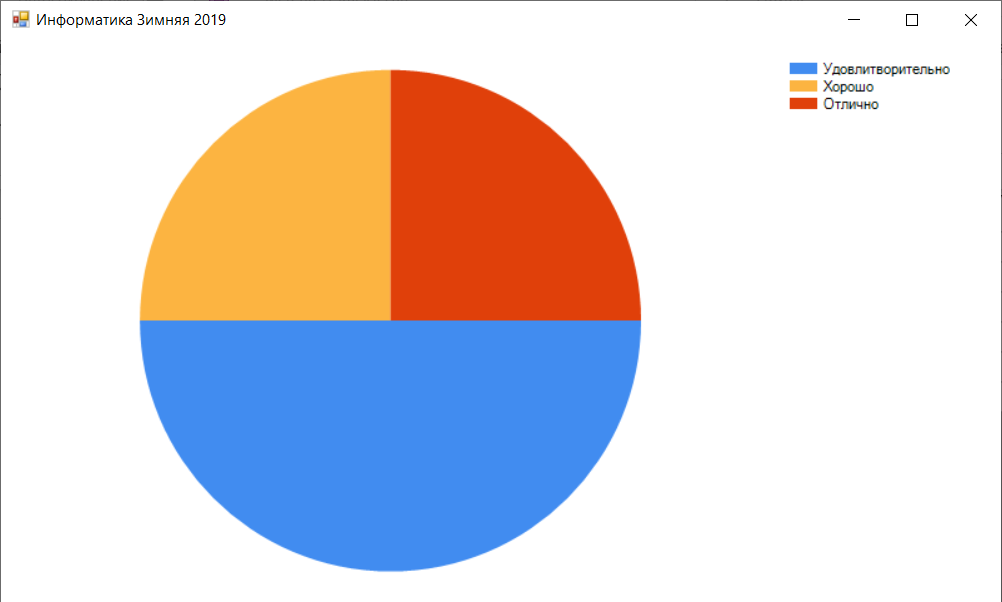


Рисунок 6 - Круговая диаграмма с оценками по Информатике

1. **Диаграмма количества студентов по всем специальностям**

Диаграмма “Количество студентов по всем специальностям” представляет собой столбиковую диаграмму с информацией о количестве студентов по всем специальностям.

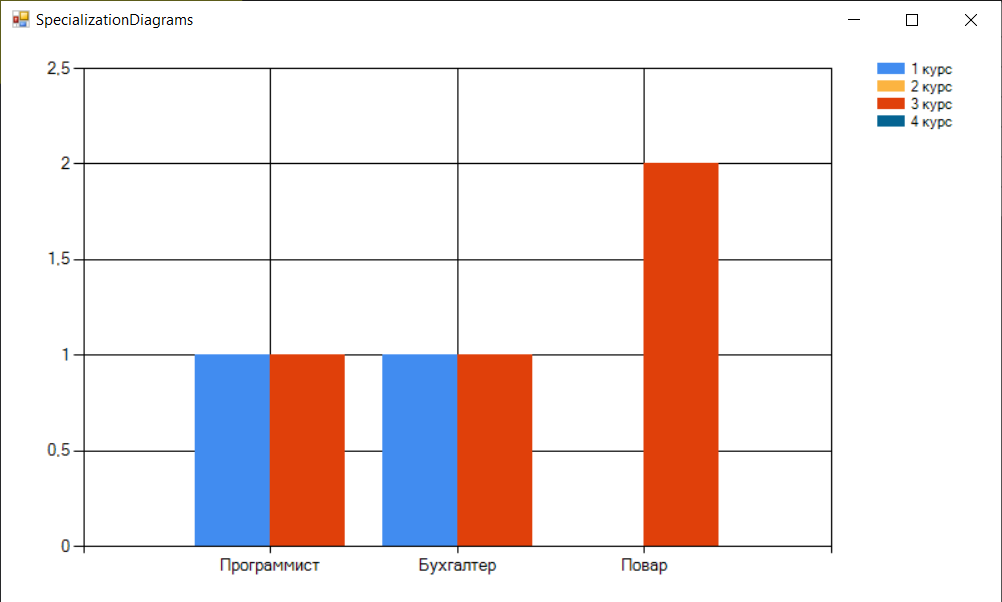


Рисунок 7 – Количество студентов по всем специальностям

**Описание тестовых сценариев**

Для проверки корректности работы методов осуществляющих выборку данных, был составлен тестовый набор (табл. 9 и табл. 10).

Таблица 9

Тестовые данные Студентов

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Фамилия** | **Имя** | **Отчество** | **Зачетка** | **Специальность** |
| Попов | Илья | Викторович | 1234 | Программист |
| Иванов | Никита | Михайлович | 5243 | Бухгалтер |
| Сидорчук | Инна | Николаевна | 5124 | Повар |
| Михеев | Игорь | Викторович | 1252 | Повар |
| Колоянов | Игорь | Сергеевич | 5114 | Программист |
| Евдокимов | Алексей | Иванович | 3232 | Бухгалтер |

Таблица 10

Тестовые данные Оценок

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Зачетка** | **Предмет** | **Сезон** | **Год** | **Оценка** |
| 1234 | Математика | Летняя | 2020 | Отлично |
| 1234 | Информатика | Зимняя | 2019 | Отлично |
| 1234 | Математика | Летняя | 2021 | Удовлитворительно |
| 1234 | Информатика | Летняя | 2021 | Удовлитворительно |
| 1234 | Русский | Зимняя | 2021 | Хорошо |
| 5243 | Математика | Летняя | 2019 | Отлично |
| 5243 | Информатика | Зимняя | 2019 | Хорошо |
| 5243 | Информатика | Летняя | 2021 | Хорошо |
| 5124 | Математика | Летняя | 2021 | Отсутсвует |
| 5124 | Информатика | Летняя | 2021 | Отсутсвует |
| 5124 | Английский | Летняя | 2021 | Отсутсвует |
| 5124 | Информатика | Зимняя | 2019 | Удовлитворительно |
| 1252 | Информатика | Зимняя | 2019 | Удовлитворительно |
| 5114 | Математика | Летняя | 2021 | Отсутсвует |
| 5114 | Информатика | Летняя | 2021 | Отсутсвует |
| 5114 | Русский | Летняя | 2021 | Отсутсвует |
| 3232 | Русский | Летняя | 2021 | Отлично |
| 3232 | Философия | Летняя | 2021 | Отлично |

Для проверки корректности работы метода *tablesFiller*, на тестовом наборе данных, был вызван метод со следующими параметрами streamReader = “Потоковой считыватель, который привязан к какому-либо файлу”. Результат вызова метода представлен на рис. 8 и рис. 9.

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 8 - заполненная таблица со студентами после вызова метода tableFiller

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 9 - заполненная таблица с результатами сессий после вызова метода tableFiller

Для проверки корректности работы метода *tableSaveFileFiller*, на тестовом наборе данных, был вызван метод со следующими параметрами streamWriter = “Экземпляр класса для сохранения данных в файл, привязанный к какому-либо файлу”. Результат вызова метода представлен на рис. 10.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 10 - файл с данными, который был создан с помощью метода tableSaveFileFiller

Для проверки корректности работы метода *debtors\_Click*, на тестовом наборе данных, был вызван метод. Результат вызова метода представлен на рис. 11.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 11 - диалоговое окно с фамилиями студентов, имеющих больше 2 долгов в последней сессии

Для проверки корректности работы метода *buttonUpperStep\_Click*, на тестовом наборе данных, был вызван метод. Результат вызова метода представлен на рис. 12.

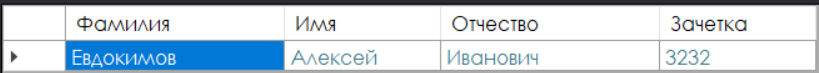


Рисунок 12 - таблица с ФИО и № зачетки со студентами претендующими на повышенную стипендию.

Для проверки корректности работы метода *graphSetter*, на тестовом наборе данных, был вызван метод со следующими параметрами data = [0,0,5,0,0,5,4,6], counters = [0,0,1,0,0,1,1,2], student = “Попов”. Результат вызова метода представлен на рис. 13.

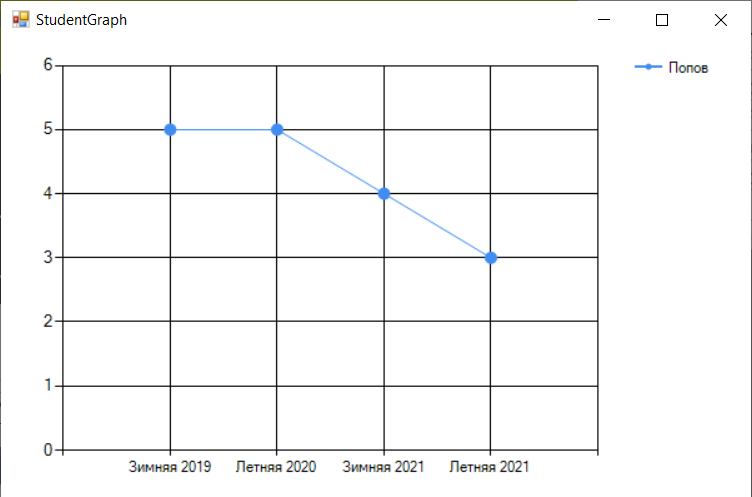


Рисунок 13 - график изменения среднего балла

Для проверки корректности работы метода *chartSetter*, на тестовом наборе данных, был вызван метод с параметрами sessionValues = [2,1,1], session = “Зимняя 2019”. Результат вызова метода представлен на рис. 14.

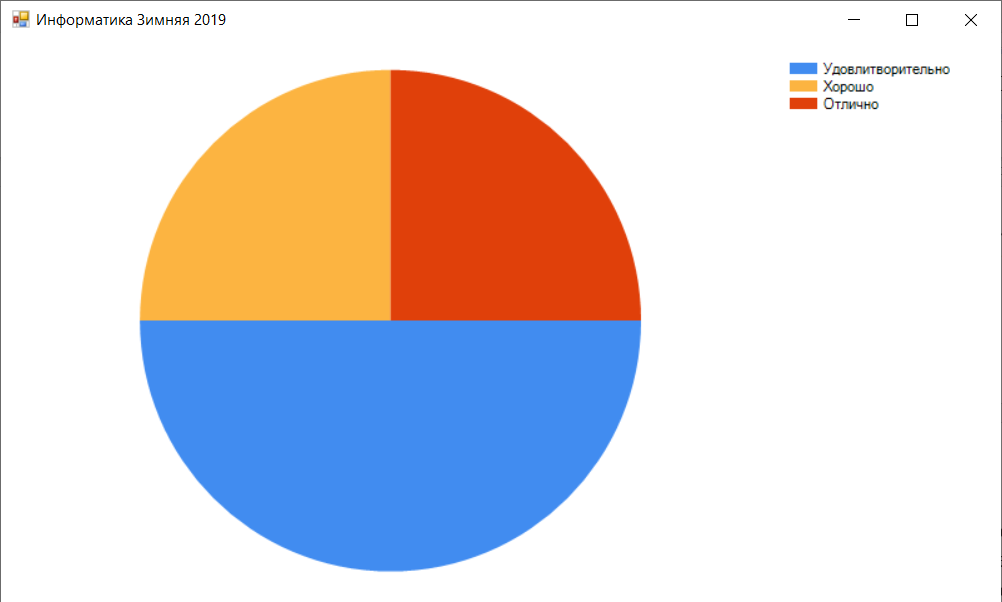


Рисунок 14 - круговая диаграмма с кол-вом оценок по информатике в выбранной сессии

Для проверки корректности работы метода *diagrammSetter*, на тестовом наборе данных, был вызван метод с параметрами allSpecializations = [“Программист”, “Бухгалтер”, “Повар”], specCounter = [0,0,0][1,1,0][0,0,0][1,1,2], allStudents = [3,3,3,3,1,1]. Результат вызова метода представлен на рис. 15.

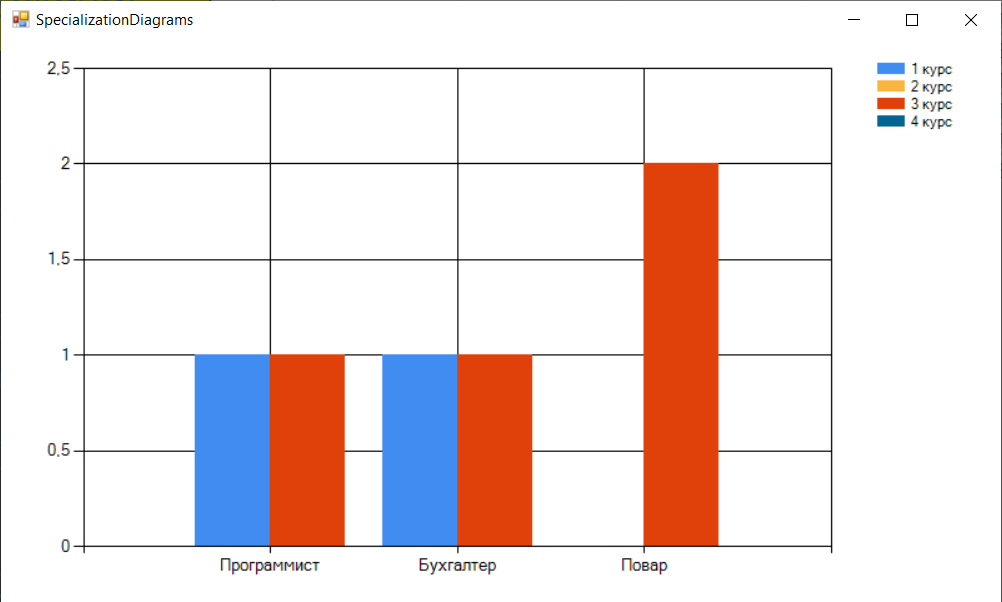


Рисунок 15 - столбчатая диаграмма с кол-вом студентов

**Список использованных источников**

1. ГОСТ 19.701-90 (ИСО 5807-85). Единая система программной документации. Схемы алгоритмов, программ, данных и систем. Условные обозначения и правила выполнения.
2. Ахо А.В., Хопкрофт Д., Джеффри Д.У.Структуры данных и алгоритмы. М.: Издательский дом «Вильямс», 2003. 384 с.
3. Вирт Н. Алгоритмы и структуры данных. М.: ДМК Пресс, 2010. 272 с.
4. Гагарина Л.Г., Колдаев В.Д. Алгоритмы и структуры данных. М.: Финансы и статистика, 2009. 304 с.
5. Гроздева С.И., Терзиева Т.Ж. Визуализация методов сортировки массива [Электронный ресурс]. URL:

<http://www.iiorao.ru/iio/pages/izdat/ison/publication/ison_2011/num_5_2011/%C3%F0%EE%E7%E4%E5%E2%20%D1.%C8.,%20%D2%E5%F0%E7%E8%E5%E2%E0%20%D2.%C6..pdf> (дата обращения: 02.09.2013)

1. Зеленяк О.П. Практикум программирования на Turbo Pascal. Задачи, алгоритмы, решения. М.: ДМК Пресс, 2007.
2. Иванова Г.С. Технология программирования. М. КноРус, 2013. 336 с. (или более ранние издания).
3. Качала В.В. Программирование на языке Турбо-Паскаль. Мурманск: МГАРФ (МГТУ), 1993. 90 с.
4. Мануйлов В.Г. Разработка программного обеспечения на Паскале. М.: ПРИОР, 1996. 240 с.
5. Методы программирования. Учебное пособие / Н.И. Минакова, Е.С. Невская, Г.А. Угольницкий, А.А. Чекулаева, М.И. Чердынцева. М.: Вузовская книга, 2000. 280 с.
6. Новичков В.С., Парфилова Н.И., Пылькин А.Н. Алгоритмизация и программирование на Турбо Паскале. М. : Горячая линия-Телеком, 2005. 438 с.
7. Павловская Т.А. Паскаль. Программирование на языках высокого уровня. СПб.: Питер, 2010. 464 с.
8. Потопахин В.В. Искусство алгоритмизации. М.: ДМК Пресс, 2011. 320 с.
9. Фаронов В.В. Турбо Паскаль 7.0. Начальный курс: учебное пособие. М.: КНОРУС, 2006. 576 с.