**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

федеральное государственное бюджетное образовательное учреждение высшего образования

**«УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Кафедра «Измерительно-вычислительные комплексы»

# Лабораторная работа

**По дисциплине «Алгоритмы и структуры данных»**

**Тема** «Объектно – ориентированная программа с графическим интерфейсом»

|  |  |
| --- | --- |
| **Подп. и** |  |
| **Инв** |  |
| **Вза** |  |
| **Подп. и** |  |
| **Инв.** |  |

# ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

**Исполнитель**:

студент гр. ИСТбд-21

Тройкин Степан

Александрович

2024

**Введение**

Настоящее техническое задание распространяется на разработку программы для взаимодействия с кредитными договорами с использованием графического интерфейса. Пользователь может загружать кредитные договоры из файла, сегментировать список договоров по менеджерам и по сумме и визуализировать данные функции в форме круговых диаграмм.

**Основания для разработки:**

Задание для лабораторной работы по учебному плану направления «Информационные системы и технологии».

**Функциональное назначение:**

1. load\_contracts **-** загрузка данных кредитных договоров из файла.
2. segment\_by\_amount – сегментирование кредитных договоров по сумме.
3. segment\_by\_manager - сегментирование кредитных договоров по менеджерам.
4. plot\_pie\_chart – Создание круговой диаграммы.

**Структура программы**

**Классы:**

1. CreditContract: Доп. Класс описаны основные элементы.
2. ContractManager: класс содержит методы для загрузки координат из файла, добавление треугольников на холст и проверки пересечений.
3. App: класс содержащий визуализацию программы и ее интерфейс.

**Стадии и этапы разработки**

Разработка проекта включала следующие этапы:

1. Определение задач и требований: установление ключевых целей и формулирование функциональных составляющих для проекта.
2. Архитектурное проектирование: определение необходимых классов, методов для реализации программы.
3. Кодирование: разработка программного кода для инициализации кредитных договоров, сегментации и создание круговых диаграмм.
4. Тестирование: проверка программы на наличие ошибок и их исправление в соответствии с поставленными задачами.
5. Оптимизация интерфейса: создание удобного диалогового окна, понятного пользователю.
6. Документация и сопровождение: разработка подробной документации, описывающей принцип работы программы.

**Тестирование**

1. Подготовка тестовых данных: создание набора файлов со списком кредитных договоров, включая некорректные значения.
2. Модульное тестирование: проверка каждого метода класса ContractManager по-отдельности.
3. Функциональное тестирование: проверка обработки событий, работы кнопок и их соответствие назначенным функциям.
4. Интеграционное тестирование: проверка взаимодействия классов ContractManager с App и графическим интерфейсом.