**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**Факультет прикладной математики-процессов управления**

**Программа бакалавриата**

**“Большие данные и распределенная цифровая платформа”**

**ОТЧЕТ**

**по лабораторной работе №1**

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

**на тему «Генерация** **датасета»**

**Вариант – 3**

**Студент гр. 23Б15-пу**

**Беляева А.П.**

**Преподаватель**

**Дик А.Г.**

**Санкт-Петербург**

**2024 г.**

Оглавление

1. [Цель работы 3](#_Toc177901232)
2. [Описание задачи (формализация задачи) 3](#_Toc177901233)
3. [Теоретическая часть 4](#_Toc177901234)
4. [Основные шаги программы](#_Toc177901235) 4
5. [Блок схема программы](#_Toc177901236) 5
6. [Описание программы](#_Toc177901237) 6
7. [Рекомендации пользователя](#_Toc177901238) 7
8. [Рекомендации программиста](#_Toc177901239) 7
9. [Исходный код программы](#_Toc177901240) 7
10. [Контрольный пример](#_Toc177901241) 8
11. [Вывод](#_Toc177901242) 10
12. [Источники 1](#_Toc177901243)0

**Цель работы**

Целью данной работы является разработка программы для генерации датасета, имитирующего покупку железнодорожных билетов. Датасет должен включать уникальные данные пассажиров, поездов, а также их поездок с различными параметрами, например, ФИО, паспортные данные, направления поездок, время отправления и прибытия, выбор места и вагона, а также параметры оплаты.

**Описание задачи (формализация задачи)**

Задача состоит в создании датасета для железнодорожных билетов со следующими требованиями:

1. **ФИО**: Славянские имена и фамилии.
2. **Паспортные данные**: Русские паспортные данные с уникальными значениями.
3. **Откуда и куда**: Города в РФ, с различием отправной и конечной точек.
4. **Дата отъезда и приезда**: Случайные даты в пределах года.
5. **Рейс**: Номера поездов в заданных диапазонах.
6. **Выбор вагона и места**: Учет классов вагонов и типов мест.
7. **Стоимость**: В зависимости от длины маршрута и типа вагона.
8. Карта оплаты: Генерация карт с повторением не больше пяти раз и возможностью настраивать вероятность к какому банку и платежной системе принадлежит карта.

**Теоретическая часть**

Ограничения:

* Количество строк в датасете — минимум 50 000.
* ФИО пассажиров только славянские.
* Уникальность паспортных данных.
* Разные точки отправки и прибытия только по территории РФ.
* Вагоны и места генерируются по заданным типам.
* Веса банков и платежных систем определяются пользователем и должны в сумме каждый давать 100.
* Логика выбора карт оплаты с ограничением на 5 повторов.

**Основные шаги программы**

1. Запуск программы.
2. Запуск интерфейса.
3. Пользователь вводит веса банков и платежных систем.
4. Генерация строк.
5. Формирование .csv файла.

# Блок схема программы

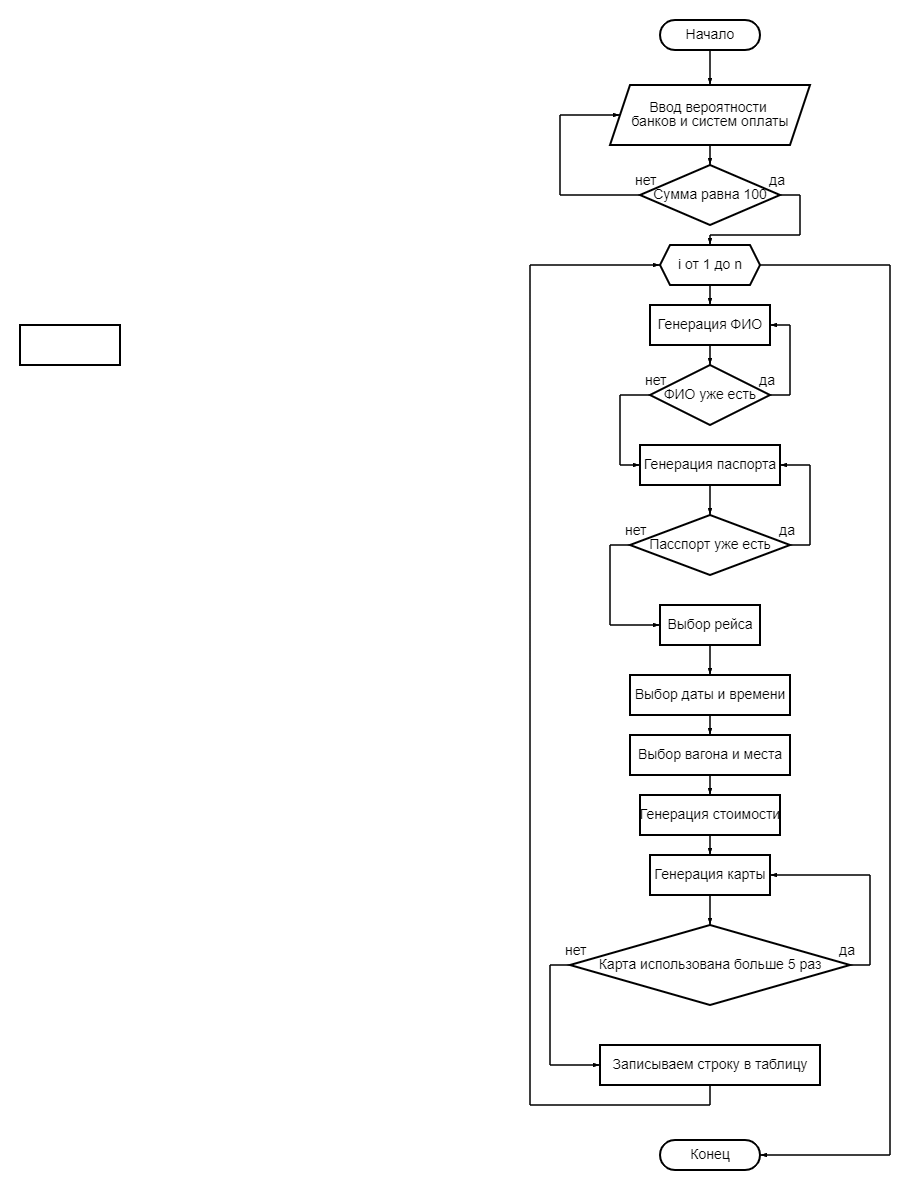


Рис 1. Блок-схема программы

# Описание программы

Программная реализация написана на языке Python 3.12 с использованием следующих библиотек: faker [[1]](#_Источники), random [[2]](#_Источники), pandas[[3]](#_Источники) и tkinter[[4]](#_Источники). В процессе разработки программы использовались 11 функций:

Таблица 1. main.py

|  |  |  |
| --- | --- | --- |
| Функция | Описание | Возвращаемое значение |
| generate\_fio | Генерация случайного ФИО | str |
| generate\_passport | Генерация паспорта | str |
| generate\_route | Генерация рейса | tuple |
| generate\_dates | Генерация дат | tuple |
| generate\_train | Генерация номера поезда | int |
| generate\_wagon\_and\_seat | Генерация вагона и места | str |
| generate\_price | Генерация стоимости | int |
| generate\_payment\_card | Генерация карты оплаты | str |
| generate\_dataset | Составление таблицы | None |
| start\_interface | Запуск интерфейса | None |
| on\_submit | Обработка выборы вероятностей банков | None |

# Рекомендации пользователя

Для использования программы необходимо открыть файл main.py в среде разработки, нажать на кнопку запуска программы и в показавшемся окошке указать параметры генерации, такие как вероятности использования различных банков и платёжных систем. Рекомендуется использовать системные требования с достаточно большим объёмом памяти, так как датасет может достигать больших размеров (50 000 строк и более).

# Рекомендации программиста

Для запуска программы убедитесь, что у вас установлен Python и необходимые библиотеки, такие как faker [[1]](#_Источники), random [[2]](#_Источники), pandas[[3]](#_Источники) и tkinter[[4]](#_Источники). Следует поддерживать актуальные версии этих библиотек. Код можно запустить из среды разработки, либо через консоль.

# Исходный код программы

[**https://github.com/hysterria/Data\_Gen.git**](https://github.com/hysterria/Data_Gen.git)

# Контрольный пример

1. Запуск программы

Для запуска программы используйте файл **main.py.** Программа будет отвечать за генерацию билетов на основе заданных данных о платежных системах и банках.

2. Ввод весов платежных систем и банков

После запуска программы пользователю предложено ввести **веса для платежных систем** и **веса для банков** (Рис. 2). Веса определяют вероятность выбора той или иной платежной системы или банка при генерации билета.

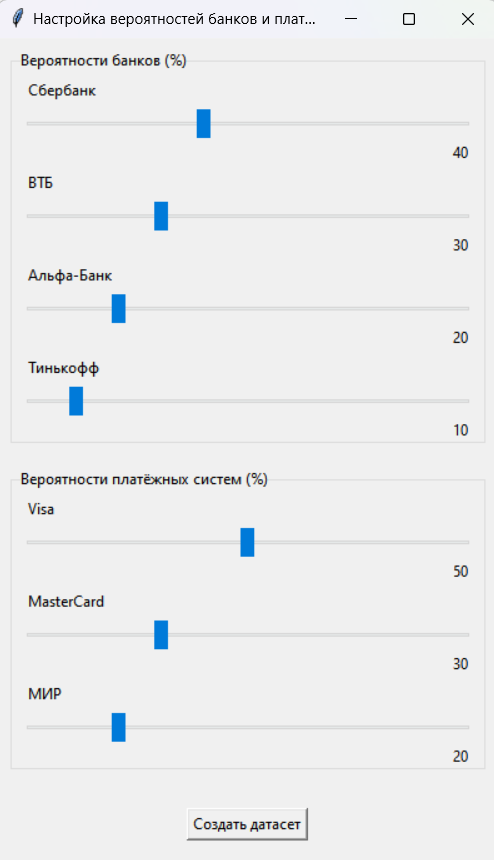


Рис 2. пример ввода весов платежных систем

3. Генерация билетов

После ввода вероятностей программа приступает к генерации таблицы данных, используя введенные параметры, затем сгенерированный датасет сохраняется в train\_tickets\_dataset.cvs.

.

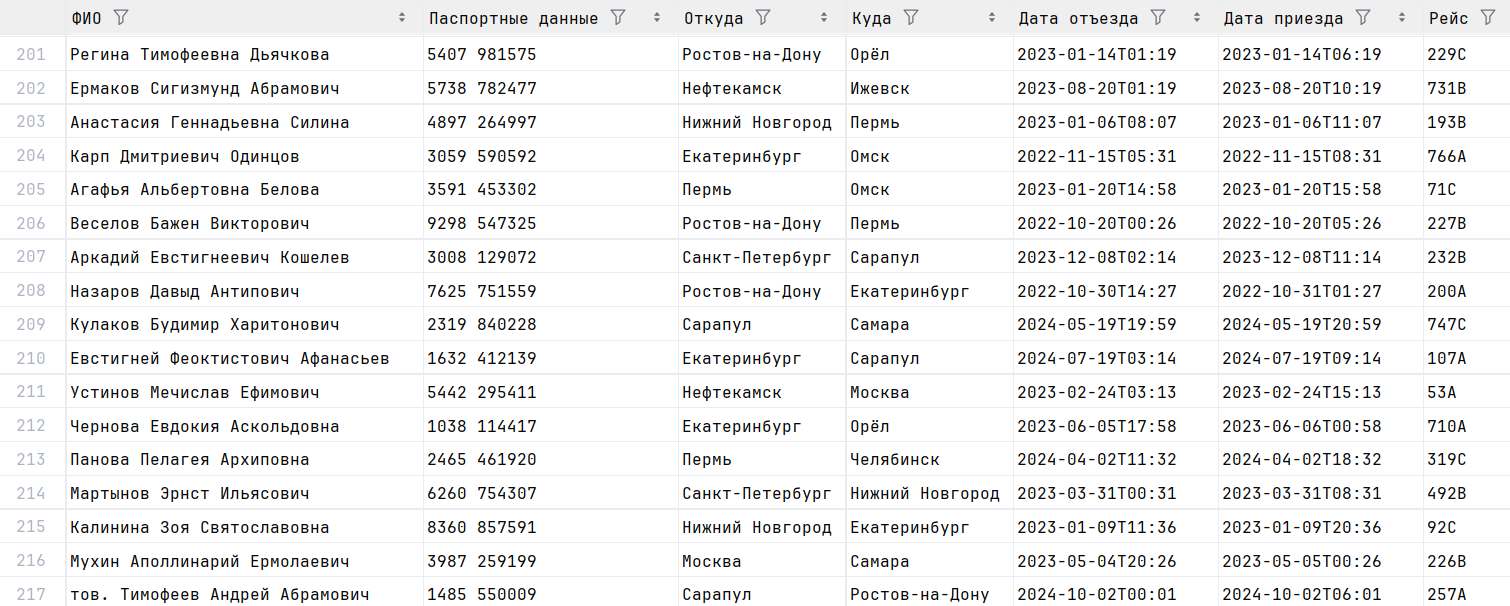


Рис 7. пример датасета билетов

# Вывод

Разработанный проект позволяет автоматизировать процесс генерации большого датасета для имитации покупки железнодорожных билетов. Программа гибкая, легко настраиваемая, и позволяет создавать данные для различных типов поездов, вагонов и маршрутов, что может быть полезно для анализа, тестирования и моделирования различных сценариев.

# Источники

1. Faker - Faker’s documentation // Faker URL: <https://faker.readthedocs.io/en/master/>.
2. Random - Generate pseudo-random numbers // Python URL: <https://docs.python.org/3/library/random.html>.
3. Pandas – pandas documentation // Pandas URL: <https://pandas.pydata.org/>.
4. Tkinter – tkinter documentation // Tkinter URL: <https://metanit.com/python/tkinter/>