

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 – 2

**Студент гр. 23Б15-пу
Сериков К.Г.**

**Преподаватель
Дик А.Г.**

Санкт-Петербург

2024 г.

Оглавление

1. Цель работы.....	3
2. Описание задачи (формализация задачи).....	3
3. Теоретическая часть.....	4
4. Основные шаги программы.....	5
5. Блок схема программы.....	6
6. Описание программы.....	7
7. Рекомендации пользователя.....	8
8. Рекомендации программиста.....	8
9. Исходный код программы.....	8
10. Контрольный пример.....	9
11. Вывод.....	11
12. Источники.....	11

Цель работы

Целью лабораторной работы является разработка системы генерации датасета для списка визитов к врачу с учетом определенных требований и условий. Датасет должен включать личные данные пациентов, информацию о симптомах, анализах и специалистах, а также данные о банковских картах.

Описание задачи (формализация задачи)

Задача состоит в создании датасета для списка визитов к врачу со следующими требованиями:

1. **ФИО:** славянские имена и фамилии.
2. **Количество строк датасета:** не меньше 50.000.
3. **Паспортные данные:** русские, белорусские и казахские паспортные данные с уникальными значениями.
4. **СНИЛС:** уникальное значение, привязан к клиенту (ФИО и паспортным данным).
5. **Симптомы:** "словарь" минимум из 5000 симптомов. Могут быть комбинации не более чем из 10 симптомов.
6. **Выбор врача:** "словарь" должен состоять минимум из 50 врачей.
7. **Дата посещения врача:** повторное посещение врача возможно только спустя 24 часа после выдачи анализов.
8. **Анализы:** "словарь" должен состоять минимум из 250 анализов. Могут быть комбинации не более чем из 5 анализов.
9. **Дата получения анализов:** через 24-72 часа после посещения.
10. **Стоимость:** свободный вариант генерации данных в рублях.
11. **Карта оплаты:** генерация карт с возможностью многократного использования с повторением не больше пяти раз и возможностью настраивать вероятность к какому банку и платежной системе принадлежит карта.

Теоретическая часть

Для создания датасета использованы несколько программных модулей:

1. `main.py`: Главный файл, отвечает за ввод данных, вызывает функции генерации из файла `functions.py`, создает файл `dataset.csv` и выводит туда данные.
2. `functions.py`: Содержит все функции для генерации данных (`generate_name`, `generate_passport`, `generate_snils`, `generate_card`, `generate_unique_card`, `generate_doctors_data`, `generate_analyses_date`, `generate_price`, `generate_visit_date`).
3. `names_data.py`: Содержит данные для генерации ФИО (мужские и женские имена, фамилии и отчества).
4. `doctors_data.py`: Содержит данные для генерации врачей, симптомов и анализов. У каждого врача есть свои подходящие симптомы и анализы.
5. `cards_data.py`: Содержит данные для генерации карт (список банков, BIN-кодов и платежных систем).

Ограничения:

- Количество строк в датасете ограничивается вводом пользователя, но минимальное количество сгенерированных строк будет 50000.
- ФИО пациентов только славянские.
- Паспортные данные уникальные и могут быть только российские, белорусские и казахские.
- Уникальность СНИЛС.
- Веса банков и платежных систем определяются пользователем.
- Одной и той же банковской картой можно платить не более пяти раз.

Основные шаги программы

- 1) Запуск программы (main.py):
- 2) Пользователь вводит количество людей, веса банков и платежных систем.
- 3) Запускается генерация словаря уникальных пациентов.
 - a) Генерация ФИО.
 - b) Генерация паспортных данных.
 - c) Генерация СНИЛС.
 - d) Генерация карт с заданным распределением.
- 4) Сбор всех данных
 - a) Случайный выбор уникального клиента.
 - b) Генерация врача.
 - i) Генерация симптомов для соответствующего врача.
 - ii) Генерация анализов для соответствующего врача.
 - c) Генерация даты визита и даты получения анализов.
 - d) Генерация стоимости.
- 5) Запись данных в ячейки.
- 6) Запись данных в файл dataset.csv.

Блок схема программы

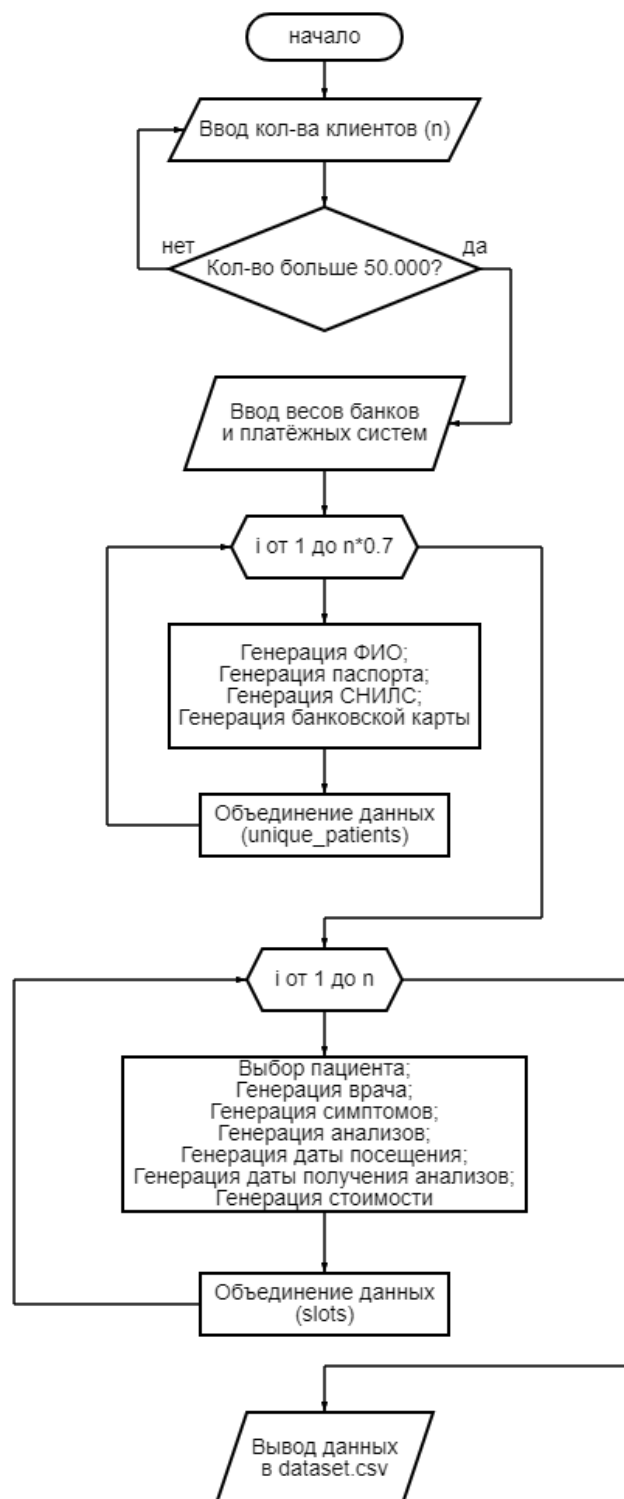


Рис 1. Блок-схема основной программы

Описание программы

Программная реализация написана на языке Python 3.11.9 с использованием следующих библиотек: datetime [1], random [2] и csv [3]. Программа нацелена на генерацию данных для списка визитов к врачу. В процессе разработки программы использовались 5 файлов и 9 функций, каждая из которых имеет чётко определённое назначение:

Таблица 1. functions.py

Функция	Описание	Возвращаемое значение
generate_name	Генерация ФИО.	str
generate_passport	Генерация паспортных данных.	str
generate_snils	Генерация СНИЛС.	str
generate_doctors_data	Генерация врача, симптомов и анализов.	list
generate_visit_date	Генерация даты и времени визита врача.	datetime
generate_analyses_date	Генерация даты получения анализов.	datetime
generate_cards	Генерация банка, системы платежей и номера карты.	list
generate_unique_card	Проверка ограничение на использование карты не более 5 раз.	list
generate_price	Генерация стоимости.	str

Рекомендации пользователя

Для запуска программы убедитесь, что у вас установлен Python. Код можно запустить в среде разработки или через командную строку, используя консоль для настройки параметров и генерации данных. Также убедитесь, что все файлы программы находятся в одной директории для корректного выполнения. Запуск программы производится через файл `main.py`, который автоматически генерирует список пациентов поликлиники в файл `dataset.csv`. Важно периодически проверять корректность данных перед генерацией походов. Если вы хотите использовать собственные файлы с данными, убедитесь в корректности структуры и заголовков: `names_data`, `doctors_data`, `cards_data`. Также настройте веса для банков и платежных систем согласно вашим требованиям, убедившись, что веса в сумме больше нуля.

Рекомендации программиста

Поддерживайте актуальную версию Python для обеспечения работоспособности программы на современных системах. Уделяйте внимание четкому именованию переменных и функций. Регулярно проводите тестирование программы на различных входных данных, чтобы убедиться в её надежности и корректности.

Исходный код программы

<https://github.com/romplle/spbu-algorithms-and-data-structures>

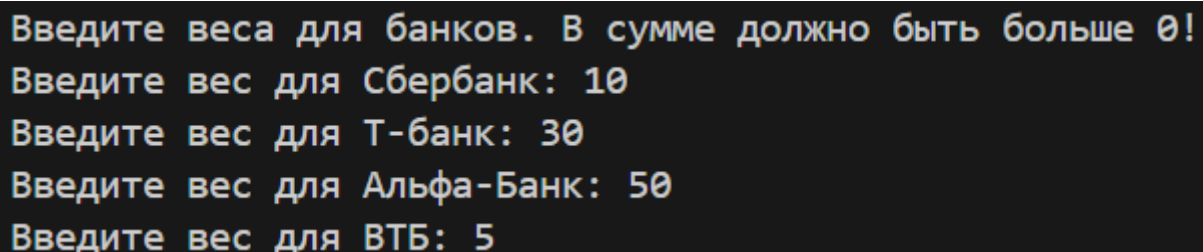
Контрольный пример

1. Запуск программы

Для запуска программы используйте файл `main.py`. Программа будет отвечать за генерацию визитов к врачам на основе заданных данных о банках и платежных системах.

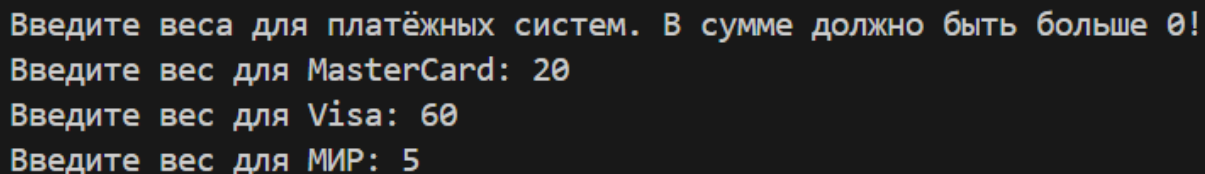
2. Ввод количества людей и весов платежных систем и банков

После запуска программы пользователю предложено ввести количество клиентов и веса для банков (Рис. 2) и платежных систем (Рис. 3). Веса определяют вероятность выбора того или иного банка или платежной системы.



```
Введите веса для банков. В сумме должно быть больше 0!  
Введите вес для Сбербанк: 10  
Введите вес для Т-банк: 30  
Введите вес для Альфа-Банк: 50  
Введите вес для ВТБ: 5
```

Рис 2. пример ввода весов платежных систем

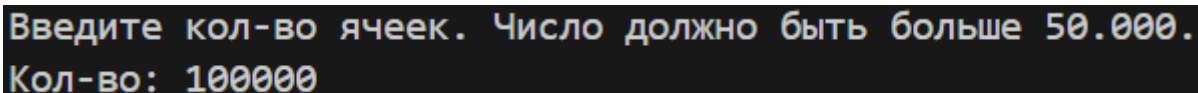


```
Введите веса для платёжных систем. В сумме должно быть больше 0!  
Введите вес для MasterCard: 20  
Введите вес для Visa: 60  
Введите вес для МИР: 5
```

Рис 3. пример ввода весов банков

3. Ввод количества пациентов

После успешной генерации пациентов пользователю предлагается ввести количество строк в датасете (Рис. 4). Минимальное количество, которое можно сгенерировать, составляет 50.000 (Рис. 5).



```
Введите кол-во ячеек. Число должно быть больше 50.000.  
Кол-во: 100000
```

Рис 4. пример ввода количества билетов

```
Введите кол-во ячеек. Число должно быть больше 50.000.  
Кол-во: 49999  
Ошибка! Введите число больше чем 49.999
```

Рис 5. пример ввода количества билетов меньшего 50000

4. Генерация пациентов

После ввода количеств весов банков и платёжных систем программа приступает к генерации пациентов и их сохранении в dataset.csv.

Вывод

В рамках данной работы были исследованы принципы генерации синтетических данных, применительно к моделированию посещений людей к врачу. Разработан алгоритм, который учитывает особенности врачей, симптомов и анализов. Было реализовано программное обеспечение для автоматической генерации датасета, включающего такие данные, как личные данные пассажиров, информация о симптомах, анализах и платежных системах. Программа позволяет настраивать параметры генерации банковских карт оплаты, обеспечивая соответствие заданным требованиям и реалистичность получаемого датасета.

Источники

1. Datetime — The datetime module supplies classes for manipulating dates and times. // Python URL: [datetime — Basic date and time types — Python 3.12.6 documentation](#) (дата обращения: 25.09.2024).
2. random — Generate pseudo-random numbers // Python URL: <https://docs.python.org/3/library/random.html> (дата обращения: 25.09.2024).
3. csv — CSV File Reading and Writing // Python URL: <https://docs.python.org/3/library/csv.html> (дата обращения: 25.09.2024).