**МИНОБРНАУКИ РОССИИ**

**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**

**Кубякин Н.А.**

**Преподаватель**

**Дик А.Г.**

**Санкт-Петербург**

**2024 г.**

***Оглавление***

[1. Цель работы 3](#__RefHeading___Toc4888_848729285)

2. Описание задачи (формализация задачи) 3

[3. Теоретическая часть 3](#__RefHeading___Toc4890_848729285)

[4. Основные шаги программы 3](#__RefHeading___Toc4892_848729285)

[5. Блок-схема программы 4](#__RefHeading___Toc4894_848729285)

[6. Описание программы 10](#__RefHeading___Toc4896_848729285)

[7. Рекомендации пользователя 11](#__RefHeading___Toc4898_848729285)

[8. Рекомендации программиста 12](#__RefHeading___Toc12522_3894689155)

[9. Исходный код программы 13](#__RefHeading___Toc12524_3894689155)

[10. Контрольный пример 14](#__RefHeading___Toc4900_848729285)

11. Вывод 15

12. Источники 16

# Цель работы

Цель данной лабораторной работы заключается в создании программы с графическим интерфейсом для автоматической генерации датасета, содержащего информацию о пациентах, посетивших частную поликлинику. Программа также предоставляет возможность задания вероятностей для генерации различных полей данных.

# Задачи

1. Изучить методы генерации данных.
2. Выбрать методы генерации, подходящие под ограничения.
3. Создать словари для генерации.
4. Добавить графический интерфейс.
5. Добавить возможность менять вероятности генерации.
6. Протестировать программу.

# Теоретическая часть

При разработке алгоритма программы для генерации датасета следует учитывать следующие условия: Формат паспорта включает в себя серию из 4 цифр, где первые 2 цифры указывают на номер региона, а последующие 2 цифры - год выдачи. Номер паспорта представляет собой счетчик для обеспечения уникальности при совпадении серий. СНИЛС состоит из 11 цифр, где последние 2 цифры являются хэш-суммой остальных цифр, остальные 9 цифр - счетчик для обеспечения уникальности номеров. Даты обращения пациента и дата результатов анализов должны соответствовать рабочим дням в рабочее время, причем дата результатов должна быть следующим рабочим днем после даты обращения. Банковская карта состоит из 16 цифр, где первая цифра определяет номер платежной системы, следующие 5 цифр - номер банка в этой системе, а оставшиеся цифры - счетчик. Количество возможных комбинаций симптомов должно быть не менее 5000, а количество анализов - не менее 250.

# Алгоритм генерации

С помощью интерфейса пользователь указывает, в каком процентном соотношении встречаются банковские карты и платежные системы. Программа начинает свою работу только при условии, что общая вероятность всех событий составляет ровно сто процентов.

Генерируется пол пациента, затем его ФИО (случайно берутся из словарей).

Генерируется серия паспорта, по ней берется номер (счетчик) и увеличивается на 1.

Генерируется СНИЛС – первые 9 цифр – счетчик, увеличивающийся на 1. 2 последние цифры генерируются случайно.

Генерируются симптомы пациента – набор размером 7 из 15 возможных вариантов. Количество возможных симптомов – сочетания из 15 по 7 (всего 6435 различных варианта)

Генерируется проводящий прием врач.

Генерируются назначаемые анализы. Алгоритм генерации аналогичен алгоритму генерации симптомов.

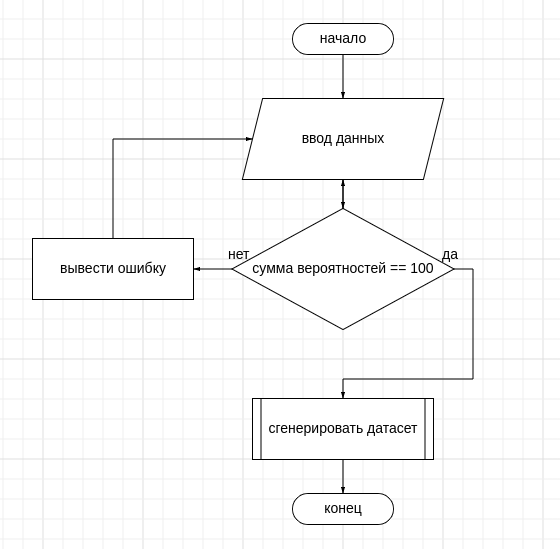
Генерируется дата и время приема, а также дата и время посещения врача. Для генерации случайной даты выбирается случайное количество дней до текущего момента. Если выбранная дата является выходным днем, из нее вычитается 2 дня для того, чтобы избежать попадания в выходной день. Время выбирается случайно из диапазона от 8 до 17 часов.

Генерируется дата получения результатов анализов. Для определения даты получения результатов анализов применяется следующий алгоритм: сначала осуществляется сдвиг даты обращения на один рабочий день вперед. Если новая дата попадает на субботу (выходной), производится дополнительный сдвиг на два рабочих дня вперед, чтобы получить дату понедельника. Таким образом, гарантируется, что дата получения результатов анализов попадает на следующий рабочий день после даты обращения, избегая выходных дней.

Генерируется стоимость анализов (шаг цены 100).

Все полученные данные записываются в файл.

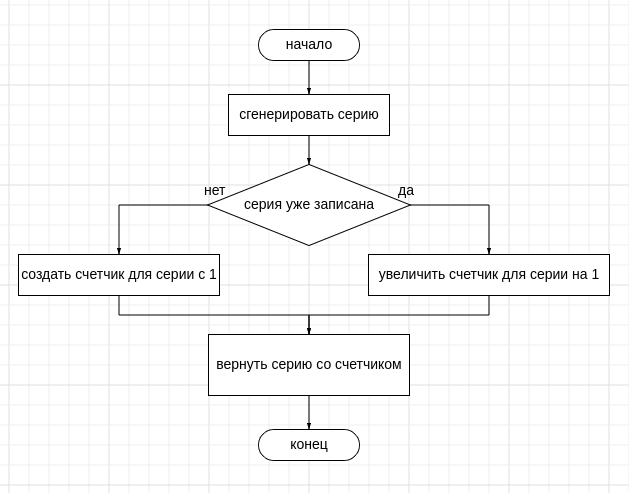
На рисунках 1-7 представлена блок-схема алгоритма.



*Рис. 4.1 Блок-схема основной программы.*



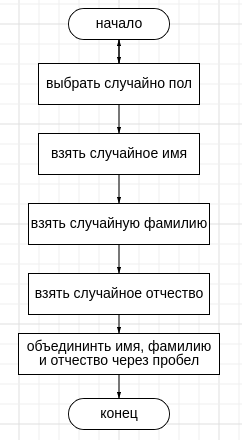
*Рис. 4.2 Блок-схема генерации датасета.*



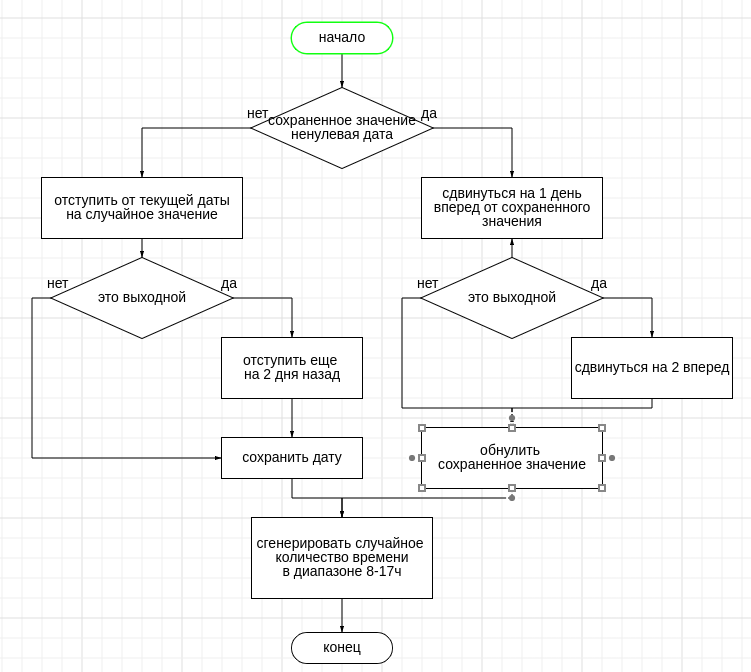
*Рис. 4.3 Блок-схема метода generate класса PassportGenerator*



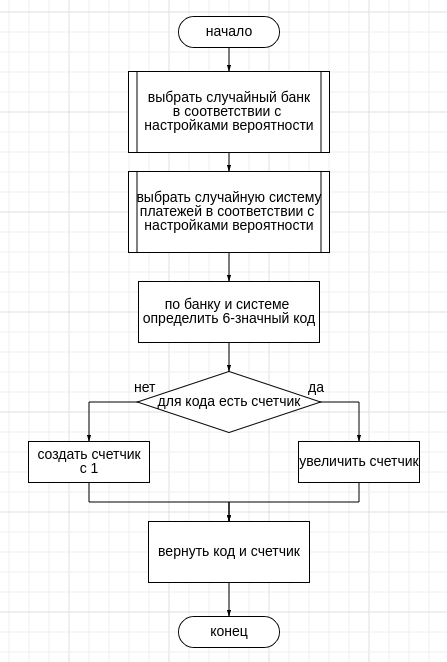
*Рис. 4.4 Блок-схема метода generate класса SnilsGenerator*



*Рис. 4.5 Блок-схема метода generate класса NamesGenerator*



*Рис. 4.6 Блок-схема метода generate класса DatetimeGenerator*



*Рис. 4.7 Блок-схема метода generate класса CardGenerator*

# Описание программы

Программа реализована на языке python 3.10 с использованием зависимостей: datetime, tkinter.

В программе используются 6 классов и 1 функция для генерации данных.

Таблица 5.1 Описание классов

|  |  |  |
| --- | --- | --- |
|  | Имя класса | Назначение |
|  | PassportGenerator | Генерирует случайную серию, от нее берется номер, являющийся счетчиком. Хранит счетчики для каждой серии. |
|  | SnilsGenerator | Генерирует СНИЛС - первые 9 цифр – счетчик, 2 последние случайные. Хранит счетчик СНИЛСов. |
|  | NamesGenerator | Генерирует ФИО. Каждый элемент берется случайно из хранящейся соответствующей базы. |
|  | DatetimeGenerator | Генерирует дату и время. Дата всегда – будний день. При первом запуске - случайная дата. При втором запуске – следующий день от предыдущей генерации. Случайное рабочее время. |
|  | SamplesGenerator | Генерирует набор симптомов или анлизов. Хранит соответствующие базы. |
|  | CardGenerator | Учитывая вероятности генерирует банк, систему, берет 6-значный ключ. Остальное – счетчик. |

Таблица 5.2 Описание функций

|  |  |  |
| --- | --- | --- |
| Имя функции | Параметры функции | Назначение |
| compute | n, out\_file, banks\_p, systems\_p | Генерирует n записей, сохраняет в файл out\_file. |

# Рекомендации пользователя

Для запуска программы необходимо установить python версии не ниже 3.10, а так же зависимости tkinker, datetime.

В каталоге «dictionary» с программой необходимо иметь специальные файлы словари. В директории должно быть 10 текстовых документов:

1. analyzis.txt - файл, содержащий словарь назначаемых анализов.
2. card\_keys.txt - файл, содержащий словарь кодов банков в каждой платежной системе.
3. doctors.txt - файл, содержащий словарь врачей, работающих в поликлинике и проводящих приемы.
4. symptoms.txt – файл, содержащий словарь симптомов, с которыми обращается пациент
5. names\_1.txt – файл, содержащий словарь славянских мужских имен.
6. names\_2.txt – файл, содержащий словарь славянских женских имен.
7. lastnames\_1.txt – файл, содержащий словарь славянских мужских фамилий.
8. lastnames\_2.txt – файл, содержащий словарь славянских женских фамилий.
9. patronymics\_1.txt – файл, содержащий словарь славянских мужских отчеств.
10. patronymics\_2.txt – файл, содержащий словарь славянских женских отчеств.

После запуска программы требуется:

1. Выбрать вероятности для генерации карт оплаты: платежных систем и банков. Если в одной из характеристик суммарная вероятность не дотянет или превысит 100, появится уведомление об ошибке.
2. В поле «Count» необходимо ввести одно целое число – количество строк генерации.
3. Для генерации необходимо заполнить все поля, нажать «Compute» дождаться уведомления об окончании. Файл с датасетом будет сохранен как значение поля «Output filename».

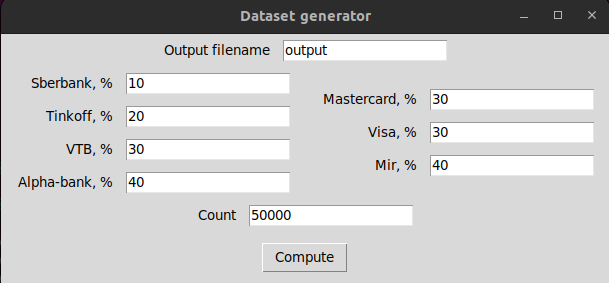
# Рекомендации программиста

Для запуска программы необходима 64-битная операционная система Windows, Linux или macOS. Для работы с кодом необходима среда разработки, совместимая с python 3.1 и библиотеки datetime и tkinter.

Исходный код программы и необходимые текстовые файлы доступны по ссылке: <https://github.com/v131v/algo_labs/tree/main/lab1>

# Контрольный пример

В данном разделе представлен пример, демонстрирующий работу программы.

Рис. 8.1 Ввод данных в интерфейс программы

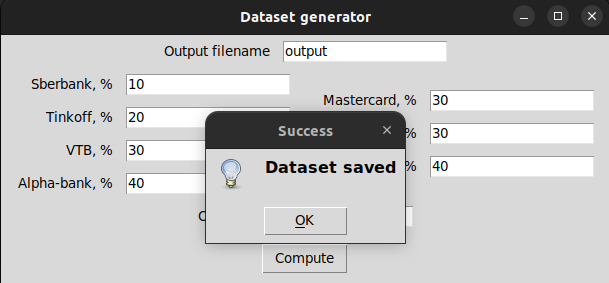


Рис. 8.2 Уведомление об успешной генерации

# Вывод

В результате выполнения данной работы разработана программа генерации датасета с синтетическими данными платной поликлиники.