**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](#_Toc178797423)

[2. Описание задачи (формализация задачи) 3](#_Toc178797424)

[3. Теоретическая часть 4](#_Toc178797431)

[4. Основные шаги программы 5](#_Toc178797432)

[5. Блок-схема программы 7](#_Toc178797433)

[6. Описание программы 14](#_Toc178797442)

[7. Рекомендации пользователя 15](#_Toc178797443)

[8. Рекомендации программиста 16](#_Toc178797444)

[9. Исходный код программы 17](#_Toc178797445)

[10. Контрольный пример 17](#_Toc178797447)

[11. Вывод 18](#_Toc178797448)

[12. Источники 18](#_Toc178797450)

# Цель работы

Цель лабораторной работы заключается в разработке 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.

# Описание задачи (формализация задачи)

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

# ФИО: Славянские имена и фамилии.

# Паспортные данные: Российские паспортные данные с уникальными значениями.

# СНИЛС: Уникальные числовые значения.

# Симптомы, анализы, врачи: Значения из словарей с определённым количеством конечных вариантов.

# Дата посещения и получения результатов: в формате гггг-мм-ддTчч:мм+ЧП (напр. 2020-01-23Т09:00+03:00).

1. **Карта оплаты**: Генерация карт с возможностью многократного использования и возможностью настраивать вероятность к какому банку и платежной системе принадлежит карта.
2. **Стоимость**: В рублях.
3. **Количество строк датасета**: Не меньше 50000.

# 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 зависят от остальных 9 цифр - счетчика для обеспечения уникальности номеров.

Формула:

1. перемножить первые 9 чисел на их положение начиная с 1 с конца и сложить.
2. если результат от 1 до 99, две цифры в конце равны результату, начиная с нуля если меньше 10.
3. если результат равен 100 или 101, последние две цифры 00.
4. если результат больше 101, взять остаток от деления суммы на 101, а затем повторить с пункта 2).

* Даты обращения пациента и дата результатов анализов должны соответствовать рабочим дням в рабочее время, причем дата результатов должна быть как минимум 24 часа 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 для уникальности.
* Количество возможных комбинаций симптомов должно быть не менее 5000, а количество анализов - не менее 250.

# Основные шаги программы

1. 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.
2. Генерируется пол пациента, затем его ФИО (случайно берутся из словарей).
3. Генерируется серия паспорта, по ней случайно генерируется номер и в случае повтора увеличивается на 1.
4. Генерируется СНИЛС – первые 9 цифр – счетчик, увеличивающийся на 1. 2 последние цифры высчитываются по вышеуказанной формуле.
5. Генерируется случайное число от 0 до 49 включительно.
6. Генерируются симптомы пациента – набор размером 5 из 10 возможных вариантов. Затем из словаря выбирается специфичный симптом. Всего 6 симптомов. Количество возможных симптомов – число сочетаний без повторений 10 по 5 (базовые симптомы) умноженное на 50 (специфичные симптомы) то есть 252\*50 = 12600 вариантов.
7. По пункту 5 из словаря выбирается врач.
8. Генерируются назначаемые анализы. Алгоритм генерации аналогичен алгоритму генерации симптомов (число сочетаний без повторений из 8 по 3 умноженное на 50, 56\*50 = 2800).
9. Генерируется дата и время приема. 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.
10. Генерируется дата получения результатов анализов. Для определения даты получения результатов анализов применяется следующий алгоритм: сначала осуществляется сдвиг даты обращения на два рабочих дня вперёд чтобы гарантированно прошло как минимум 24 часа (ибо часы могут уменьшиться). 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.
11. Генерируется стоимость анализов (шаг цены 100 рублей).
12. Генерируется уникальный номер карты с учётом введённых вероятностей банков и платёжных систем.
13. Все полученные данные записываются в файл.

# Блок-схема программы

# На рисунках 1-7 представлена блок-схема алгоритма.

# 

# Рис. 5.1 Блок-схема основной программы.

# 

# Рис. 5.2 Блок-схема генерации датасета.

# 

# Рис. 5.3 Блок-схема метода генерации паспорта

# 

# Рис. 5.4 Блок-схема метода генерации СНИЛС

# 

# Рис. 5.5 Блок-схема метода генерации ФИО

# 

# Рис. 5.6 Блок-схема метода генерации дат

# 

# Рис. 5.7 Блок-схема метода генерации банковских карт

# Описание программы

Программа реализована на языке Python 3.10 с использованием библиотек random [[1]](#random), datetime [[2]](#datetime), pandas [[3]](#pandas), tkinter [[4]](#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 считаются по формуле. |
|  | NamesGenerator | Генерирует ФИО. Каждый элемент берется случайно из соответствующих словарей. |
|  | DatetimeGenerator | Генерирует дату и время. Дата всегда – будний день. При первом запуске - случайная дата. При втором запуске – несколько дней от предыдущей генерации. |
|  | SamplesGenerator | Генерирует набор симптомов или анализов. Каждый элемент берется случайно из соответствующих словарей. |
|  | 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 выходной файл out\_file. |

# Рекомендации пользователя

Для запуска программы необходимо установить Python версии не ниже 3.10, а так же библиотеки random [[1]](#random), datetime [[2]](#datetime), pandas [[3]](#pandas), tkinter [[4]](#tkinter).

В каталоге «dictionary» с программой необходимо иметь словари. В директории должно быть 12 текстовых документов:

1. analysis.txt - файл, содержащий словарь назначаемых базовых анализов.
2. analysis\_specific.txt - файл, содержащий словарь назначаемых специфичных анализов.
3. card\_keys.txt - файл, содержащий словарь кодов банков в каждой платежной системе.
4. doctors.txt - файл, содержащий словарь врачей, работающих в поликлинике и проводящих приемы.
5. symptoms.txt – файл, содержащий словарь базовых симптомов, с которыми обращается пациент.
6. symptoms\_specific.txt – файл, содержащий словарь специфичных симптомов, с которыми обращается пациент.
7. names\_1.txt – файл, содержащий словарь славянских мужских имен.
8. names\_2.txt – файл, содержащий словарь славянских женских имен.
9. lastnames\_1.txt – файл, содержащий словарь славянских мужских фамилий.
10. lastnames\_2.txt – файл, содержащий словарь славянских женских фамилий.
11. patronymics\_1.txt – файл, содержащий словарь славянских мужских отчеств.
12. 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 достигнет 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 «Create database» дождаться уведомления об окончании. Файл с датасетом будет сохранен как значение поля «Filename» в формате .csv.

# 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 random [[1]](#random), datetime [[2]](#datetime), pandas [[3]](#pandas), tkinter [[4]](#tkinter).

# Исходный код программы

# Исходный код программы и необходимые текстовые файлы доступны по ссылке: <https://github.com/NikiTaku1/spbu_alg/tree/main/lab1>

# Контрольный пример

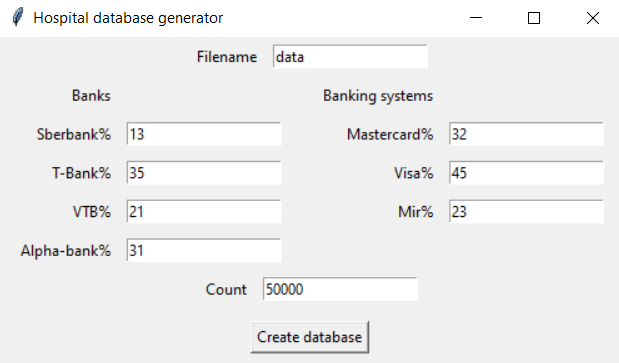
В данном разделе представлен пример, демонстрирующий работу программы.

Рис. 10.1 Ввод данных в интерфейс программы

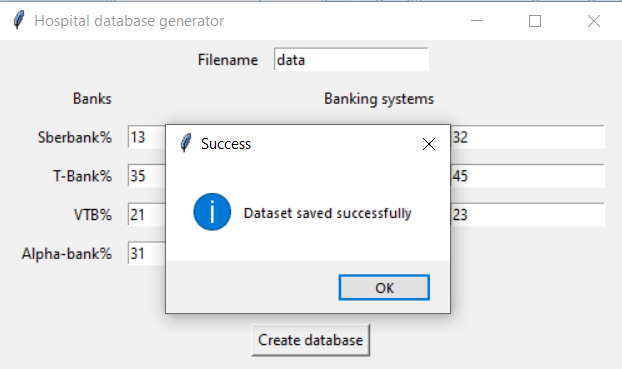


Рис. 10.2 Уведомление об успешной генерации

# Вывод

# В рамках выполнения данной работы были исследованы принципы генерации синтетических данных, применительно к моделированию базы данных платной поликлиники. Разработан алгоритм, который учитывает особенности данных пациентов, принципы генерации номеров документов, разновидности симптомов, врачей и возможных направлений на анализы. Было реализовано программное обеспечение для автоматической генерации датасета, включающего такие данные, как личные данные пациентов, информация о симптомах, врачах и анализах, датах посещения и получения результатов анализов и стоимостях услуг. Программа позволяет настраивать необходимые параметры генерации банковских карт оплаты, обеспечивая соответствие требованиям и правдоподобность итогового датасета.

# Источники

# random — Generate pseudo-random numbers // Documentation URL: <https://docs.python.org/3/library/random.html> (дата обращения: 25.09.2024).

# datetime — Basic date and time types // Documentation URL: <https://docs.python.org/3/library/datetime.html> (дата обращения: 25.09.2024).

# pandas — pandas documentation // Documentation URL: <https://pandas.pydata.org/docs/> (дата обращения: 25.09.2024).

# tkinter — Python interface to Tcl/Tk // Documentation URL: <https://docs.python.org/3/library/tkinter.html> (дата обращения: 25.09.2024).