**МИНОБРНАУКИ РОССИИ**

**САНКТ-ПЕТЕРБУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**КАФЕДРА КОМПЬЮТЕРНОГО МОДЕЛИРОВАНИЯ И МНОГОПРОЦЕССОРНЫХ СИСТЕМ**

**ОТЧЕТ**

**по лабораторной работе**

**по дисциплине “Алгоритмы и структуры данных”**

**на тему “Генерация базы данных”**

|  |  |
| --- | --- |
| Студент | Михайлов И. С. |
| Преподаватель | Щеголева Н. Л. |

Санкт-Петербург

2022 г.

[Цель работы 2](#_Toc651666751)

[Задача 3](#_Toc1880917611)

[Алгоритм метода 4](#_Toc499140503)

[Диаграмма классов 4](#_Toc132729127)

[Описание классов 5](#_Toc902234188)

[Карта кода 6](#_Toc993697311)

[Пример сгенерированных данных 7](#_Toc683901103)

[Рекомендации пользователя 7](#_Toc186932852)

[Рекомендации программиста 8](#_Toc1592991213)

[Вывод 9](#_Toc2052537104)

[Ссылки 9](#_Toc998608392)

# Цель работы

Сгенерировать базу данных по определенным требованиям.

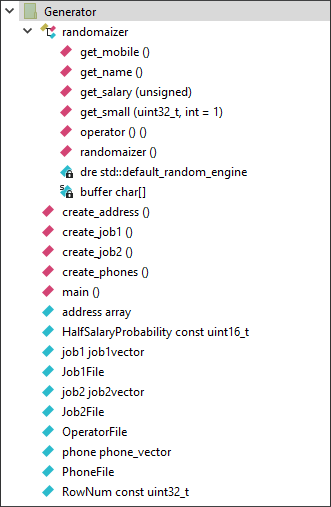
# Задача

1. Сгенерировать датасет в котором будут следующие наборы свойств:
   1. ФИО - Иванов Иван Иванович
   2. Номер телефона - “+79001234567”
   3. Адрес работы - Университетская наб., д. 7
   4. Должность - Менеджер
   5. З/П - 50 000 руб
2. Дополнительная информация по каждому свойству (Санкт-Петербург):
   1. ФИО - свободный вариант генерации данных.
   2. Номер телефона - возможность настраивать сам датасет. Например, чтоб больше номеров Мегафона было, чем МТС, а остальных поровну и возможность настраивать вероятность внутреннего регионального кода поставщика услуг (Мегафон, МТС и тд)
   3. Адрес работы - ”Словарь” по которому будет генерироваться улица + задать определенный (реальный) промежуток номеров домов
   4. Должность - ”Словарь” по которому будет генерироваться должность в определенной пропорции, которую возможно изменять в будущем (Ген. директор - 1, Начальник - 10%, Обслуживающий персонал (Охранник, Уборщица, Комендант и тд) - 20%, Бухгалтер - 5%, Менеджер - 25%, Основные профессии вашей фирмы (Программисты, Инженеры, Медики и тд) - 35% и Секретарь - 5%)
   5. З/П - средняя по данной специальности в городе и сделать вероятность, которая сделает, сколько людей получают пол. ставки. (Менеджер - 50 000 руб, Менеджер - 25 000 рублей)
3. Ограничения датасета:
   1. Всего строк в датасете - минимум 50 000.
   2. ФИО - словарь должен состоять только из славянских ФИО
   3. Номер телефона - все операторы и вариации региональных кодов по Санкт-Петербургу.
   4. Минимальное значение людей по адресу - 50 человек.
   5. З/П - разновидностей по формуле = (Набор профессий + (Набор профессий/2)) Пример если профессий у нас 15, то вариаций по З/П следует, 15+(15/2)= 18. так как округляем до целого числа частное.

# Алгоритм метода

Программа считывает информацию о мобильных операторах и должностях из соответствующих файлов. Все остальные базы данных уже содержатся в самой программе. Пока не сгенерировано нужное число записей, в таблицу вносятся новые фирмы. Перед началом генерации фирмы определяется число ее работников как рандомное число от 50 до 500. Также производится проверка - если для следующей фирмы остается менее 50 строк, то данная фирма становится последней, и к ней добавляются оставшиеся строки. Для каждой фирмы сначала записываются работники, у которых указано точное количество (Ген. директор - 1), затем оставшиеся строки распределяются между работниками, количество которых указано в процентах.

# Диаграмма классов

Рис. 1

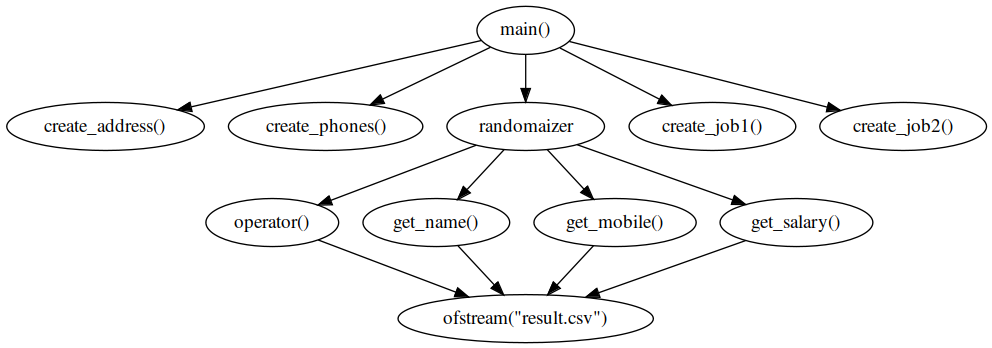
# Описание классов

В программе используется один класс - randomaizer. Он содержит в себе стандартный генератор случайных чисел и с помощью него генерирует значения для полей таблицы. Описание его методов представлено в таблице 1.

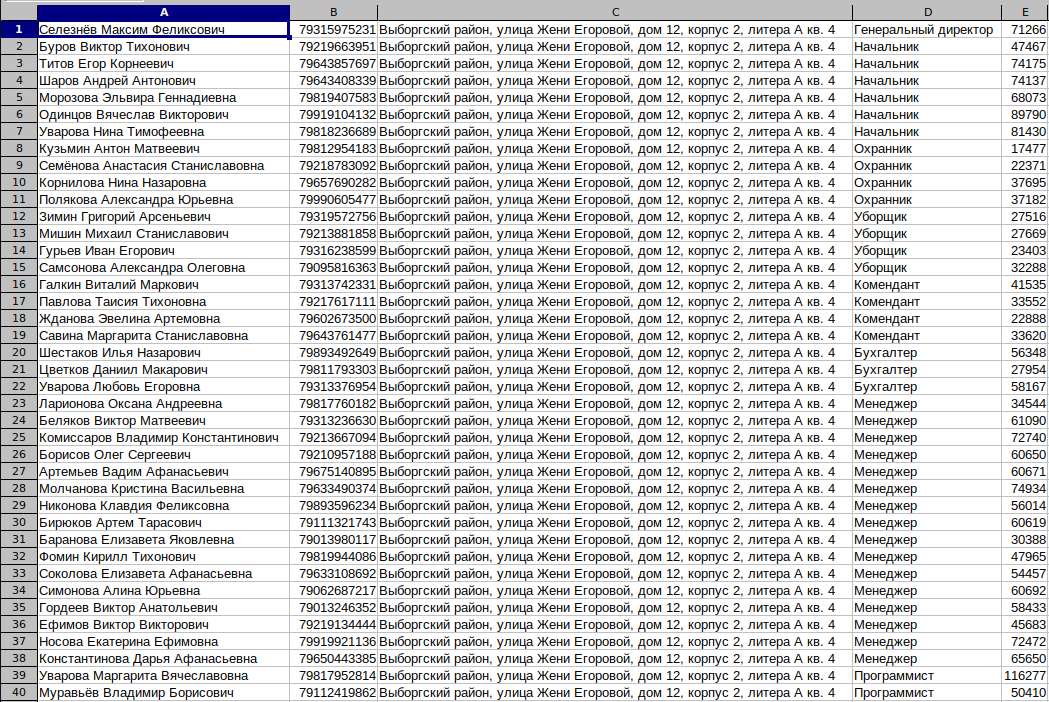
|  |  |
| --- | --- |
| **Сигнатура** | **Описание** |
| **randomaizer**() | Конструктор |
| uint32\_t operator**()**() | Возвращает случайное число (вызывает в себе аналогичный метод стандартного генератора) |
| uint32\_t **get\_small**(uint32\_t **range**, int **pow** = 1) | Возвращает случайное число в диапазоне [0, range), маленькие числа выпадают с большей вероятностью, чем большие. Чем больше pow, тем более неравномерно распределение. |
| const char\* **get\_name**() | Возвращает комбинацию рандомных фамилии, имени и отчества |
| uint64\_t **get\_mobile**() | Возвращает номер телефона, сгенерированный по условиям из задачи |
| unsigned **get\_salary** (unsigned **average**) | Возвращает зарплату, зная среднюю по профессии. То есть число в диапазоне [0.75\*average, 1.25\*average]. С заданной вероятностью делит зарплату на 2. |

Табл. 1

# Карта кода

Рис. 2

# Пример сгенерированных данных

Рис. 3

# Рекомендации пользователя

В папке с программой должны находится файлы с настраиваемыми параметрами генерации - operator.csv, phone.csv, job.csv, job\_proportion.csv.

Описание полей в operator.csv:

1. Имя мобильного оператора
2. Его вероятность. Сумма вероятностей может быть любой. По умолчанию в вероятности записано количество всех номеров оператора

Описание полей в phone.csv:

1. Имя оператора
2. Первый номер в диапазоне
3. Последний номер в диапазоне
4. Вероятность самого диапазона. Сумма вероятностей может быть любой. По умолчанию это количество номеров в диапазоне.

Описание полей в job.csv:

1. Название должности
2. Средняя зарплата
3. Точное количество людей в фирме с данной должностью

Описание полей в job\_proportion.csv:

1. Название должности
2. Средняя зарплата
3. Количество людей в фирме с данной должностью, пропорционально остальным

В результате работы программы появляется файл result.csv, содержащий сгенерированную базу данных.

# Рекомендации программиста

Программа собирается с помощью cmake версии не ниже 3.8 и использует с++ 20-й версии. Никакие сторонние библиотеки не нужны. Вероятность появления людей, работающих на полставки, содержится в переменной HalfSalaryProbability в файле main.cpp. Количество строк в таблице содержится в переменной RowNum в файле main.cpp.

# Вывод

В ходе выполнения данной работы удалось сделать программу, генерирующую базу данных с параметрами из входных файлов.

# Ссылки

* [futuris-tech/DatabaseGenerator at v2 (github.com)](https://github.com/futuris-tech/DatabaseGenerator/tree/v2)