Структура приложения представлена на рисунке 2.

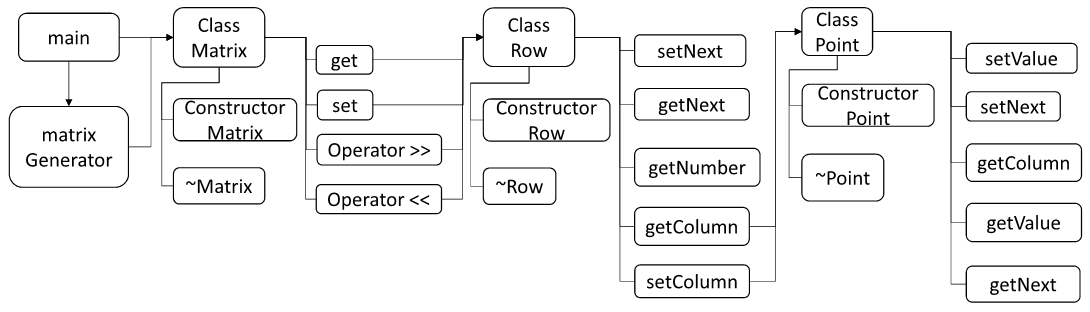


Рис. 2. Структура приложения

**Функция main**

Тип: int.

Выходные данные: не используются.

Назначение: демонстрация работы разработанной библиотеки.

Внешние эффекты: в зависимости от демонстрационных целей можно:

* Генерировать матрицу
* Сжимать матрицу (путём прямого прохода расширенной матрицы)
* Записывать сжатую матрицу в файл (сериализация) – на основе потоков.
* Считывать матрицу из файла (десериализация) – на основе потоков.
* Получать и изменять значения сжатой матрицы.

**Функция matrixGenerator**

Тип: Matrix

Выходные данные: квадратная сжатая матрица (представление на основе связанных списков)

Входные значения: поток для записи (для демонстрации используется файловый поток), разрешение матрицы, частота появления ненулевых элементов, максимальное значение ненулевого элемента.

Внешние эффекты: запись расширенной матрицы в поток. При использовании файлового потока можно открыть полученный файл в Excel.

**Класс Matrix**

Назначение: хранение, запись и вывод разреженных матриц.

Информационные поля: указатель на первую строку.

**Конструктор класса Matrix**

Тип: объект класса Matrix.

Назначение: инициализация информационных полей класса.

**Деструктор класса Matrix**

Назначение: освобождение памяти, удаление строк.

**Метод get класса Matrix**

Тип: int.

Назначение: получение элемента матрицы по координатам.

Входные данные: номер строки и номер столбца.

Выходные данные: значение в данной ячейке матрицы.

**Метод set класса Matrix**

Тип: void.

Назначение: установка значения в заданной ячейке матрицы. Если значение матрицы стало 0 или вся строка стала пустой, будет соответственно удалена ячейка или вся строка.

Входные данные: номер строки, номер столбца, новое значение.

**Оператор >> класса Matrix**

Тип: &ostream

Назначение: вывод сжатой матрицы в поток (сериализация). Если был передан файловый поток, то матрица будет записана (в сжатом виде) в файл.

Входные данные: выходной поток, матрица.

**Оператор << класса Matrix**

Тип: &istream

Назначение: чтение сфатой матрицы из потока (десериализация). Если был передан файловый поток, то матрица будет прочитана из файла.

Входные данные: поток, матрица

**Класс Row**

Назначение: хранение строки матрицы.

Информационные поля: номер строки, указатель на следующую строку.

**Конструктор класса Row**

Тип: объект класса Row.

Входные данные: номер строки.

Назначение: инициализация информационных полей класса.

**Деструктор класса Row**

Назначение: освобождение памяти, удаление строк.

**Метод setNext класса Row**

Тип: void.

Назначение: установка следующей строки.

Входные данные: указатель на строку.

**Метод getNext класса Row**

Тип: Row.

Назначение: получение следующей строки

**Метод getNumber класса Row**

Тип: int.

Назначение: получение номера строки

**Метод getColumn класса Row**

Тип: int.

Назначение: получение значения ячейки

Входные данные: номер столбца ячейки

**Метод setColumn класса Row**

Тип: boolean.

Назначение: установка значения ячейки.

Входные данные: номер столбца ячейки, значение.

Выходные данные: true, если строка не пуста.

**Класс Point**

Назначение: хранение ячейки матрицы.

Информационные поля: номер столбца, значение ячейки, указатель на следующую ячейку.

**Конструктор класса Point**

Тип: объект класса Point.

Входные данные: номер строки.

Назначение: инициализация информационных полей класса.

**Деструктор класса Point**

Назначение: освобождение памяти, удаление ячеек (вызов деструктора следующей ячейки).

**Метод setNext класса Point**

Тип: void.

Назначение: установка следующей ячейки.

Входные данные: указатель на ячейку.

**Метод getNext класса Point**

Тип: Point.

Назначение: получение следующей ячейки

**Метод getColumn класса Point**

Тип: int.

Назначение: получение номера столбца

**Метод getValue класса Point**

Тип: int.

Назначение: получение значения ячейки