**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра МО ЭВМ**

отчет

**по лабораторной работе №5**

**по дисциплине «Объектно-ориентированное программирование»**

Тема: Обработчик последовательности.

|  |  |  |
| --- | --- | --- |
| Студент гр. 7304 |  | Моторин Е.В. |
| Преподаватель |  | Размочаева Н.В |

Санкт-Петербург

2019

**Цельработы:**

Реализовать программу обработчика последовательности, которые представляет из себя параллельно последовательный граф с одним входом и одним выходом.

1. **Задача:**

Реализовать обработчик последовательности объектов стандартного (*int, string, double, char*) типа. Обработчик представляет собой ориентированный параллельно-последовательный граф с одним входом и выходом. Каждая вершина графа является функциональным элементом, выполняющим какое-то действие над объектом.

Функциональные элементы могут быть различаться по количеству входов и выходов: 1 вход – 1 выход; 1 вход – 2 выхода; 2 входа – 1 выход: 2 входа – выхода. Вход обработчика может быть элементом только с 1 входов, а выход обработчика может быть элементом только с 1 выходом. Причем, функциональные элементы с 2 входами выполняются только когда объекты поступили на оба входа. Например, могут быть следующие функциональные элементы: умножение числа на другое число n; определение более длинной строки; расчет целой части и остатка от деления числа на число n.

Такт – одна итерация работы всего обработчика. За один такт выполняется каждый функциональный элемент, передает объект дальше и принимает следующий. Таким образом, работа обработчика происходит по принципе конвейера.

При запуске программы, набор обработчиков заранее известен, но должна быть возможность построения обработчика. Также должна быть возможность замены одного функционального элемента на другой в ходе работы обработчики, причем состояние должно сохраняться.

Также необходимо предусмотреть возможность отката на один такт назад.

Минимальные требования к заданию:

1. По 2 типа обработчика для каждой комбинации входа/выхода
2. Обработчик должен работать с двумя стандартными типами
3. Обработчик должен работать с одним пользовательским типом

Дополнительное задание (выбрать минимум одно):

1. Замена одного функционального элемента набором функциональных элементов
2. Реализация обработчика с несколькими входами и выходами. Добавляется возможность обработки разных последовательностей одного типа
3. Функциональный элементы могут преобразовывать тип объектов.
4. **Ход работы:**

**2.1.** В качестве дополнительного задания был реализован GUI c использованием средств QT.

**2.2.** Был реализован паттерн адаптер для конвертации формата JSON в граф.

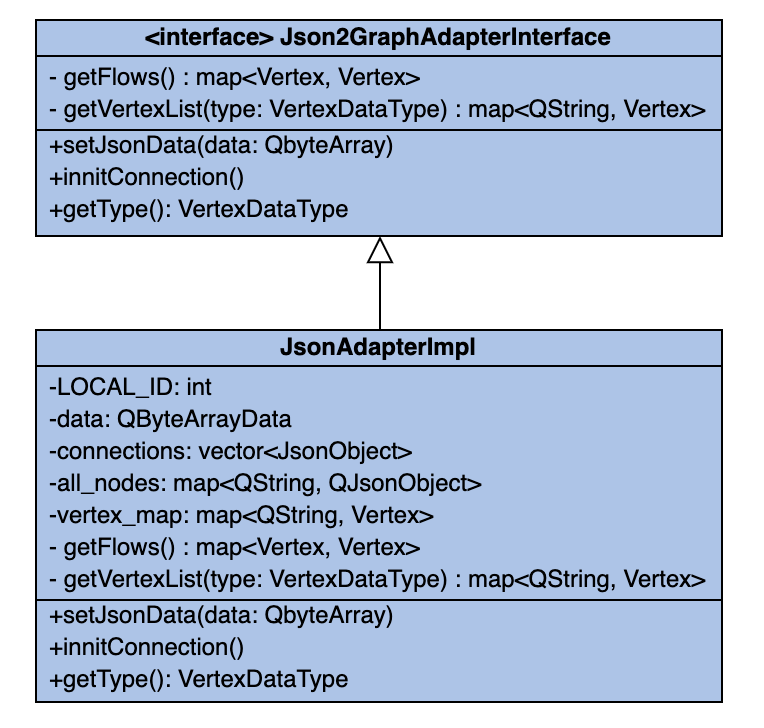


Рис. 1

**2.3.** Реализован паттерн снимок для хранения состояний графа. Снимок позволяет запоминать каждое из состояний графа и возвращаться к ним.

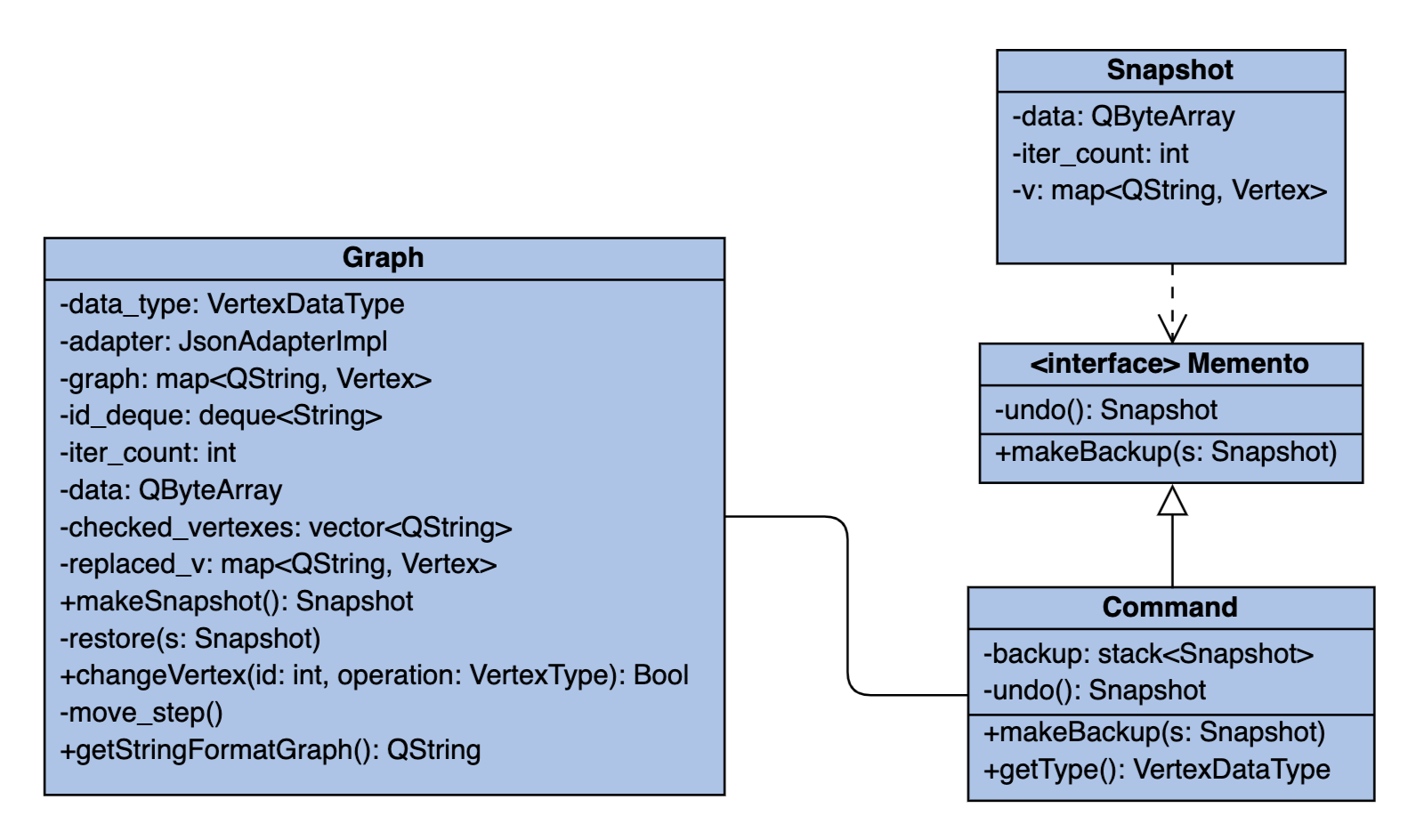


Рис. 2

**Вывод:**

Таким образов, в ходе данной лабораторной работы были изучены паттерны проектирования.