

## Паттерны проектирования ПО

### Самостоятельная работа №1

#### ЗАДАНИЕ.

Создать интерфейс *ВЕКТОР* – доступные извне методы должны обеспечивать чтение и запись значений компонентов вектора, должно присутствовать доступное извне свойство только для чтения *РАЗМЕРНОСТЬ\_ВЕКТОРА*.

Создать интерфейс *МАТРИЦА* – доступные извне методы должны обеспечивать чтение и запись значений элементов матрицы, должны присутствовать доступные извне свойства только для чтения *ЧИСЛО\_СТРОК* и *ЧИСЛО\_СТОЛБЦОВ*.

Реализовать поддержку интерфейса *ВЕКТОР* в конкретных классах *ОБЫЧНЫЙ\_ВЕКТОР* и *РАЗРЕЖЕННЫЙ\_ВЕКТОР*.

Реализовать поддержку интерфейса в абстрактном классе *НЕКОТОРАЯ\_МАТРИЦА*, от которого породить конкретные классы *ОБЫЧНАЯ\_МАТРИЦА* и *РАЗРЕЖЕННАЯ\_МАТРИЦА*. В классе *ОБЫЧНАЯ\_МАТРИЦА* для представления матрицы используйте класс *ОБЫЧНЫЙ\_ВЕКТОР*. В классе *РАЗРЕЖЕННАЯ\_МАТРИЦА* для представления матрицы используйте класс *РАЗРЕЖЕННЫЙ\_ВЕКТОР*.

Создать класс *ИНИЦИАТОР\_МАТРИЦ*, который имеет единственный статический метод *ЗАПОЛНИТЬ\_МАТРИЦУ* со следующим набором параметров: *МАТРИЦА*, число ненулевых значений в матрице, максимально возможное значение.

Создать конкретный класс *СТАТИСТИКА\_МАТРИЦЫ* с параметризованным конструктором, на вход которого подается *МАТРИЦА*. У данного класса должен быть доступен извне набор свойств только для чтения: сумма всех значений матрицы, среднее значение, максимальное значение, число ненулевых значений.

В **КЛИЕНТСКОЙ ЧАСТИ** системы создайте объекты классов *ОБЫЧНАЯ\_МАТРИЦА* и *РАЗРЕЖЕННАЯ\_МАТРИЦА*; используя класс *ИНИЦИАТОР\_МАТРИЦ* заполните их случайными значениями; выведите на консоль информацию о значениях матрицы, полученную из класса *СТАТИСТИКА\_МАТРИЦЫ*.

ПОСЛЕДНИЙ СРОК СДАЧИ: 19.04.2022

## Паттерны проектирования ПО

### Самостоятельная работа №2

#### ЗАДАНИЕ.

Разработать систему визуализации матриц. Обычная матрица (*ОБЫЧНАЯ\_МАТРИЦА*) и РАЗРЕЖЕННАЯ матрица (*РАЗРЕЖЕННАЯ\_МАТРИЦА*) должны иметь разные схемы построения изображения. Любая матрица состоит из следующих визуальных элементов: *внешней границы, содержимого ячеек*.

Изображение обычной матрицы должно формироваться так: рисуется *внешняя граница*, затем последовательно рисуются визуальные элементы для ВСЕХ ячеек (*содержимое*). Изображение разреженной матрицы должно формироваться немного иначе: рисуется *внешняя граница*, затем последовательно рисуются визуальные элементы ТОЛЬКО НЕНУЛЕВЫХ ячеек (*содержимое*). Т.е. визуальные элементы ячеек, содержащих 0, вообще не изображаются.

Под конкретной *схемой визуализации матрицы* будем понимать текстовое или графическое представление визуальных элементов матрицы (*внешней границы, содержимого ячеек*). Необходимо реализовать в системе визуализации матриц поддержку нескольких схем визуализации (обязательное использование шаблона *BRIDGE*). Необходимо реализовать в системе возможность изменения схемы визуализации матрицы в режиме выполнения кода.

Реализовать две схемы визуализации матрицы. Первая схема выводит элементы матрицы на консоль. Вторая схема отображает визуальные элементы матрицы на графический контекст. Для всех схем визуализации опционально должна задаваться возможность *отображать/скрывать границу матрицы*.

Сценарий работы **КЛИЕНТСКОЙ ЧАСТИ**. Открывается диалоговое окно, где представлены: кнопка «ГЕНЕРАЦИЯ ОБЫЧНОЙ МАТРИЦЫ», кнопка «ГЕНЕРАЦИЯ РАЗРЕЖЕННОЙ МАТРИЦЫ», графическая панель и консольная панель для визуализации матриц, кнопка выбора (checkbox) опции отображать границу матрицы.

По открытию окна все кнопки должны быть активны. Нажатие одной из кнопок «ГЕНЕРАЦИЯ...» происходит: 1) создание соответствующей матрицы-объекта, 2) заполнение матрицы случайными значениями, 3) визуализация матриц с использованием двух схем визуализации. Клик на кнопку выбора опции приводит к перерисовки матрицы с границей/без границы.

ПОСЛЕДНИЙ СРОК СДАЧИ: 19.04.2022

## Паттерны проектирования ПО

### Самостоятельная работа №3

#### ЗАДАНИЕ.

Разработать для матриц (объекты, поддерживающие интерфейс *МАТРИЦА*) механизм перенумерации строк и столбцов. В основе указанного механизма положить шаблон *DECORATOR*:

1. реализовать интерфейс *МАТРИЦА* в конкретном классе *ПЕРЕНУМЕРУЮЩИЙ\_ДЕКОРАТОР*;
2. снабдить класс *ПЕРЕНУМЕРУЮЩИЙ\_ДЕКОРАТОР* переменной-ссылкой на декорируемый объект *МАТРИЦА* и обеспечить инициализацию указанной переменной в конструкторе;
3. написать декорирующий код для всех методов *ПЕРЕНУМЕРУЮЩИЙ\_ДЕКОРАТОР*;
4. описать и реализовать пару новых **public** методов, которые обеспечивают перенумерацию строк и столбцов *ПЕРЕНУМЕРУЮЩИЙ\_ДЕКОРАТОР*.

Изменения в **КЛИЕНТСКОЙ ЧАСТИ**. Добавить в диалоговое окно: кнопку «ПЕРЕНУМЕРОВАТЬ», кнопку «ВОССТАНОВИТЬ». По нажатию на кнопку «ПЕРЕНУМЕРОВАТЬ» происходит обмен пары случайно выбранных строк и пары случайно выбранных столбцов матрицы (**декорирование**). По нажатию на кнопку «ВОССТАНОВИТЬ» происходит восстановление матриц (**раздекорирование**).

ПОСЛЕДНИЙ СРОК СДАЧИ: 19.04.2022