

ГУАП  
КАФЕДРА № 51

ПРЕПОДАВАТЕЛЬ

доцент, к.т.н.		Линский Е. М.
должность , уч. степень, звание	подпись, дата	инициалы, фамилия

## ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЫ №5 СОЗДАНИЕ ПРОГРАММЫ НА ЯЗЫКЕ JAVA

по курсу: ТЕХНОЛОГИИ ПРОГРАММИРОВАНИЯ

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. №	5022		Б.Э.Штеле
		подпись, дата	инициалы, фамилия

Санкт-Петербург 2022

# Задание

Напишите интерфейс `IMatrix` с несколькими реализациями --- `UsualMatrix` и расширяющий его `SquareMatrix` из предыдущих заданий и `SparseMatrix` для разреженных матриц. `SparseMatrix` должен быть реализован с помощью `LinkedList` (возможно, вам потребуется создать какие-то еще дополнительные классы, которые должны быть вложенными/внутренними). Все общие методы должны быть представлены в интерфейсе `IMatrix`.

Напишите программу, создающую 2 случайные матрицы размером 1000x1000 с 1000 ненулевых элементов в каждой двумя способами --- с помощью обычных и разреженных матриц. Проверьте, что сложение и умножение для разных видов матриц дает одинаковые результаты

## Инструкция

Программист-пользователь самостоятельно вызывает методы, а именно `sum(IMatrix)`-для того, чтобы сложить матрицы друг с другом, `product(IMatrix)`-для того, чтобы перемножить матрицы, метод `get(int row, int column)` чтобы получить значение в указанной строке и столбце, метод `set(int row, int column, int value)` позволяющий установить значение в выбранную строку и столбец, метод `equals(Object)` в каждом классе, для того чтобы проще сравнивать матрицы на равенство.

## Тестирование

Тест 1

```
UsualMatrix a = new UsualMatrix(100,100);
SparseMatrix g = new SparseMatrix(100, 100);
UsualMatrix b = new UsualMatrix(100,100);
SparseMatrix c = new SparseMatrix(100, 100);
int f = 0;
int k = 0;
for(int i = 0; i < a.getRow();i++){
    a.set(i, i, i);
    g.set(i, i, i);
    b.set(i, i, i);
    c.set(i, i, i);
}

a = (UsualMatrix)a.product(c);
```

```
g =(SparseMatrix) g.product(b);  
System.out.println(a.equals(g));  
Данный пример программы выведет true
```

## Тест 2

```
SparseMatrix c = new SparseMatrix(5, 5);  
c.set(3, 4, 5);  
c.set(4 4, 6);
```

После установки значений матрица будет выглядеть вот так

0 0 0 0 0

0 0 0 0 0

0 0 0 0 0

0 0 0 0 5

0 0 0 0 6