

ГУАП  
КАФЕДРА № 51

ПРЕПОДАВАТЕЛЬ

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

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №9  
СОЗДАНИЕ ПРОГРАММЫ НА ЯЗЫКЕ JAVA

по курсу: ТЕХНОЛОГИИ ПРОГРАММИРОВАНИЯ

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. №	5721		А.Е.Ковалева
		подпись, дата	инициалы, фамилия

Санкт-Петербург 2019

## Задание 9. Synchronized.

### 1. Основное задание

Написать программу, приводящую к ситуации взаимной блокировки (deadlock).

### 2. Дополнительное задание

[reader-writer]

Реализовать класс MyRandomList на основе LinkedList с тремя функциями:

void addNumber()-добавление случайного элемента,

void removeNumber()-удаление элемента со случайным индексом,

int calcZero()-подсчет числа ненулевых элементов.

Реализовать потоки двух видов:

потоки первого вида MyWriteThread добавляют или удаляют из MyRandomList случайные элементы,поток второго вида MyReadThread подсчитывает число ненулевых элементов в массиве и выводят результат на экран.

Сделать две версии класса MyRandomList с синхронизацией и без.

Показать, что при использовании версии без синхронизации возникает ошибка (необходимо подобрать число потоков)

## Инструкция

### 1. Основное задание.

Deadlock описывает ситуацию, когда два или более потоков заблокированы навсегда, ожидая друг друга. Взаимная блокировка возникает, когда нескольким потокам требуются одинаковые блокировки, но они получают их в другом порядке. Ключевое слово synchronized заставляет исполняющий поток блокировать при ожидании блокировки или монитора, связанного с указанным объектом.

При компиляции и запуске программы обнаруживается ситуация взаимоблокировки.

Программа зависнет, так как оба потока ждут, когда каждый снимет блокировку.

### 2. Дополнительное задание.

Есть файлы main.java, MyRandomList.java для версии без синхронизации и main\_Synchronized.java, MyRandomList\_Synchronized.java для версии с синхронизацией соответственно. В обоих MyRandomList прописаны методы void addNumber(), void removeNumber(), int calcZero().

Файл MyWriteThread.java общий для двух версий реализации программы. В нем потоки добавляют и удаляют случайные элементы, обращаясь к методам, прописанным в

соответствующих версиях MyRandomList , а так же выводят общее число элементов в массиве и количество удаленных элементов.

Файл MyReadThread.java тоже общий для обеих версий запуска программы. В нем прописано обращение потоков к методу подсчета числа нулевых элементов.

После запуска программы в версии без синхронизации на экран выводится сообщение об ошибке.

После запуска программы в версии с синхронизацией на экран выводятся массивы случайных элементов, число элементов, число нулевых элементов и количество удаленных элементов.

Выводится массив элементов с удаленными символами.

## Тестирование

### 1. Основное задание

```
"C:\Program Files\JetBrains\IntelliJ IDEA Community Editi
Thread-0 --- lock
Thread-1 --- lock
Thread-0   Wait another thread
Thread-1   Wait another thread
|
```

### 2. Дополнительное задание

#### 2.1 Версия без синхронизации

```
without synchronized:
Exception in thread "Thread-3" java.lang.NullPointerException
    at java.base/java.util.LinkedList$Itr.next(LinkedList.java:897)
    at MyRandomList.calcZero(MyRandomList.java:32)
    at MyReadThread.run(MyReadThread.java:23)
6 3 1 7 4 0 7 6 3 1 9 7 6 4 2 0 3 7 7 9 1 2 1 7 7 1 9 9 4 3 3 5 0 7 9 2 7 1 9 3 5 5 6 3 1 7 4 0 7 9 2 7 1 9 3 5 Exception in thread "Thread-0" java.lang.NullPointerException
    at java.base/java.util.LinkedList$Itr.next(LinkedList.java:897)
    at MyRandomList.print(MyRandomList.java:45)
    at MyWriteThread.run(MyWriteThread.java:29)
Exception in thread "Thread-1" Exception in thread "Thread-2" java.lang.NullPointerException
    at java.base/java.util.LinkedList$Itr.next(LinkedList.java:897)
    at MyRandomList.print(MyRandomList.java:45)
    at MyWriteThread.run(MyWriteThread.java:29)
java.lang.NullPointerException
    at java.base/java.util.LinkedList$Itr.next(LinkedList.java:897)
    at MyRandomList.print(MyRandomList.java:45)
    at MyWriteThread.run(MyWriteThread.java:29)
Exception in thread "main" java.lang.NullPointerException
    at java.base/java.util.LinkedList$Itr.next(LinkedList.java:897)
    at MyRandomList.print(MyRandomList.java:45)
    at main.main(main.java:18)

Process finished with exit code 1
```

## 2.2 Версия с синхронизацией

### Test 1

```
with synchronized:
7 6 5 3 6
7 6 5 3 6 6 8 6 4 4 7 8
7 6 5 3 6 6 8 6 4 4 7 8 8 7 9 8 7 3
Количество элементов: 6
число нулевых элементов: 0
Количество элементов: 18
Количество элементов: 12
число удаленных элементов:1
число удаленных элементов:1
число удаленных элементов:0
7 6 5 3 6 6 6 4 4 8 8 7 9 8 7 3

Process finished with exit code 0
```

### Test 2

```
with synchronized:
9 7 1 8 2 1 6
9 7 1 8 2 1 6 0 4 9 0 1
9 7 1 8 2 1 6 0 4 9 0 1 8 3 9 8
Количество элементов: 12
Количество элементов: 8
число нулевых элементов: 2
Количество элементов: 16
число удаленных элементов:1
число удаленных элементов:0
число удаленных элементов:1
9 7 8 1 6 0 4 9 0 1 8 3 9 8

Process finished with exit code 0
```