



Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Московский государственный технический университет  
имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)

---

ФАКУЛЬТЕТ \_\_\_\_\_ «Информатика и системы управления»

КАФЕДРА \_\_\_\_\_ «Теоретическая информатика и компьютерные технологии»

**Лабораторная работа № 4**  
**по курсу «Языки и методы программирования»**  
**«Реализация итераторов в языке Java»**

Студент группы ИУ9-21Б Горбунов А. Д.

Преподаватель Посевин Д. П.

*Москва 2023*

# 1 Задание

Во время выполнения лабораторной работы требуется разработать на языке Java один из классов, перечисленных в таблицах 1 – 7. Класс должен реализовывать интерфейс `Iterable`.

Объект разрабатываемого класса должен быть изменяемым, то есть в нём надо так или иначе предусмотреть возможность изменения внутреннего состояния. В методе `main` вспомогательного класса `Test` нужно продемонстрировать работоспособность разработанного класса.

Последовательность целых чисел с итератором по всем соседним парам чисел.

## 2 Результаты

Исходный код программы представлен в листинге 1, 2.

Листинг 1 — класс `Test`

```
1 public class Test {  
2     public static void main(String[] args) {  
3         Sequence sequence = new Sequence(1, 2, 3, 4, 5);  
4         for (int s : sequence) {  
5             System.out.print(s + " ");  
6         }  
7         System.out.println();  
8         sequence.set(2, 10);  
9         sequence.set(4, 20);  
10        for (int s : sequence) {  
11            System.out.print(s + " ");  
12        }  
13        System.out.println();  
14    }  
15 }
```

## Листинг 2 — класс Sequence

```
1 import java.util.Iterator;
2 public class Sequence implements Iterable<Integer> {
3
4     private int[] sequence;
5
6     public Sequence(int... sequence) {
7         this.sequence = sequence;
8     }
9
10    public void set(int index, int value) {
11        sequence[index] = value;
12    }
13
14    @Override
15    public Iterator<Integer> iterator() {
16        return new NeighborIterator();
17    }
18
19    private class NeighborIterator implements Iterator<Integer> {
20        private int index = 0;
21        @Override
22        public boolean hasNext() {
23            return index < sequence.length - 1;
24        }
25        @Override
26        public Integer next() {
27            int next = sequence[index] + sequence[index + 1];
28            index++;
29            return next;
30        }
31        @Override
32        public void remove() {
33            throw new UnsupportedOperationException();
34        }
35    }
36 }
```

Результат запуска представлен на рисунке 1, 2, 3.

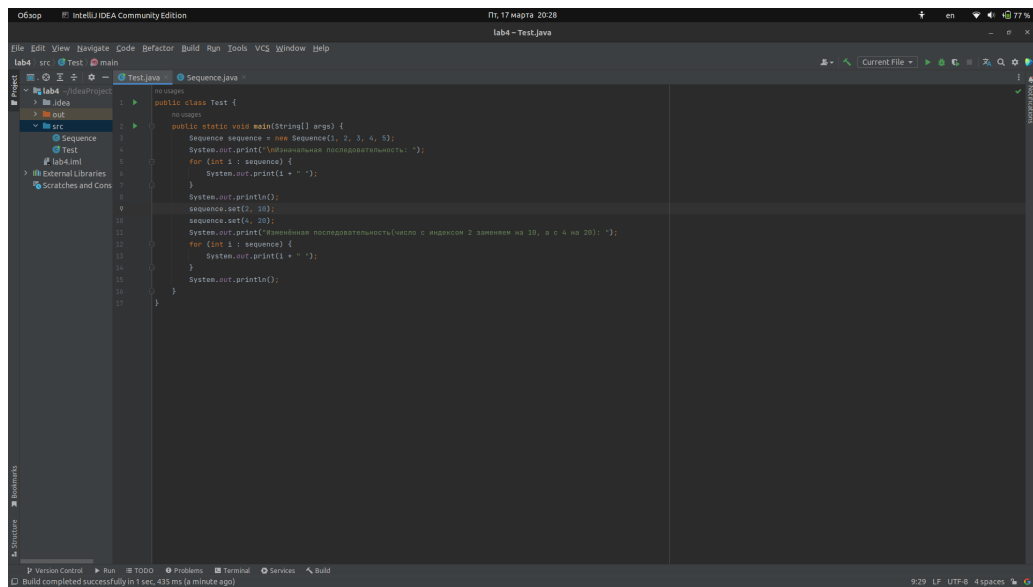


Рис. 1 — Работа программы

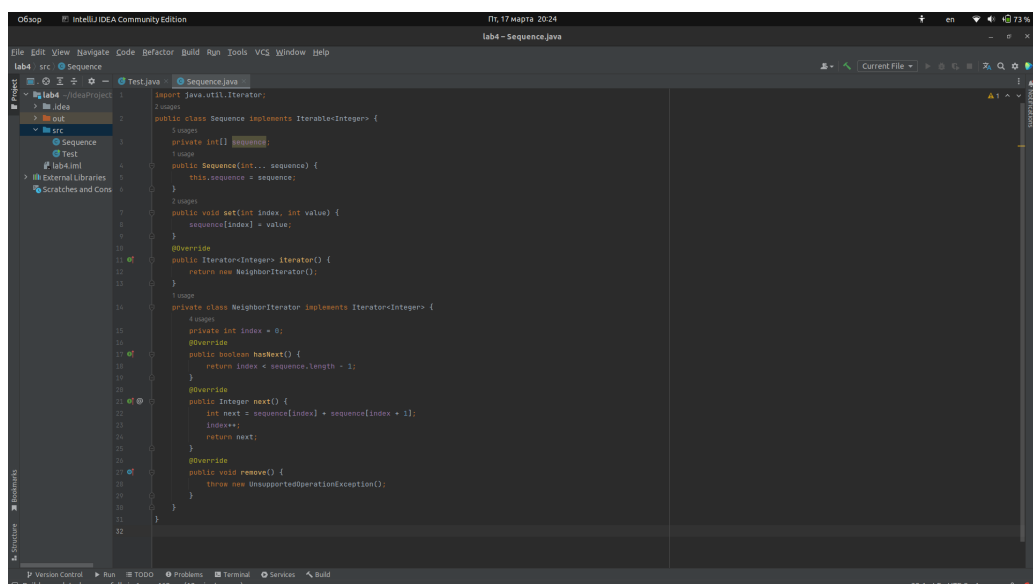


Рис. 2 — Работа программы

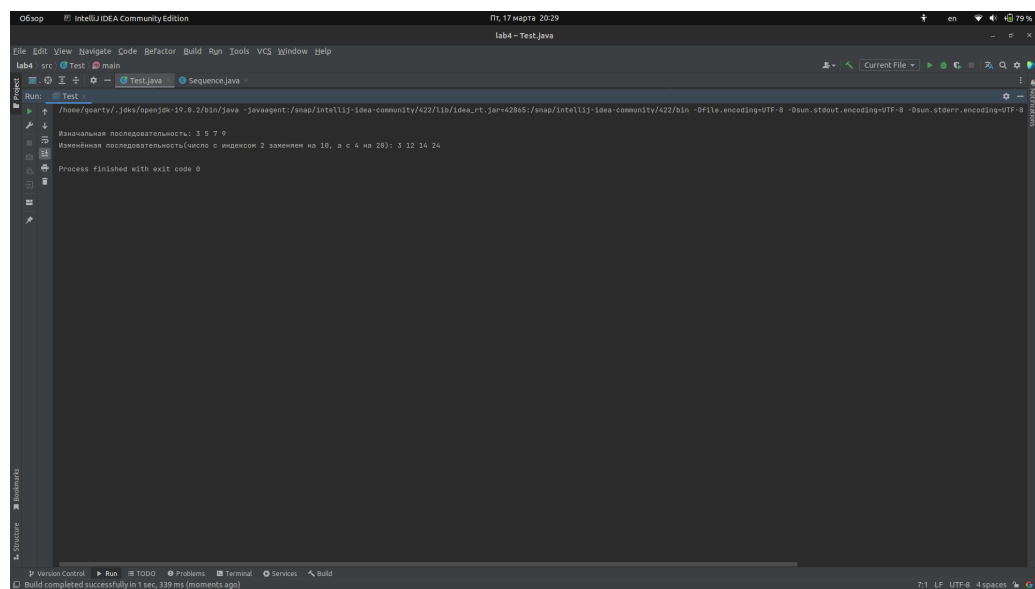


Рис. 3 — Вывод программы