



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»**  
**«Монады в языке Java. Вариант 16»**

Студент группы ИУ9-22Б Жук 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 таблице, которая приведена ниже. В каждом классе нужно реализовать по крайней мере два метода: первый метод должен возвращать `Stream`, а второй – `Optional`. Операции, выполняемые каждым методом, указаны в вариантах задания. В методе `main` вспомогательного класса `Test` нужно продемонстрировать работоспособность разработанного класса, осуществив группировку содержимого потока, возвращаемого первым методом, с помощью группирующего коллектора. В исходном коде (включая класс `Test`) запрещено использовать циклы и рекурсию.

16 Последовательность треугольников на плоскости, заданных целочисленными координатами вершин, с операциями: 1. порождение потока, содержащего прямоугольные треугольники из последовательности, гипотенуза каждого из которых превышает гипотенузу предыдущего прямоугольного треугольника последовательности; 2. вычисление максимальной площади среди площадей прямоугольных треугольников, входящих в последовательность.

## 2 Результаты

Далее для проверки поменяем порядок(1ый треугольник со 2ым)

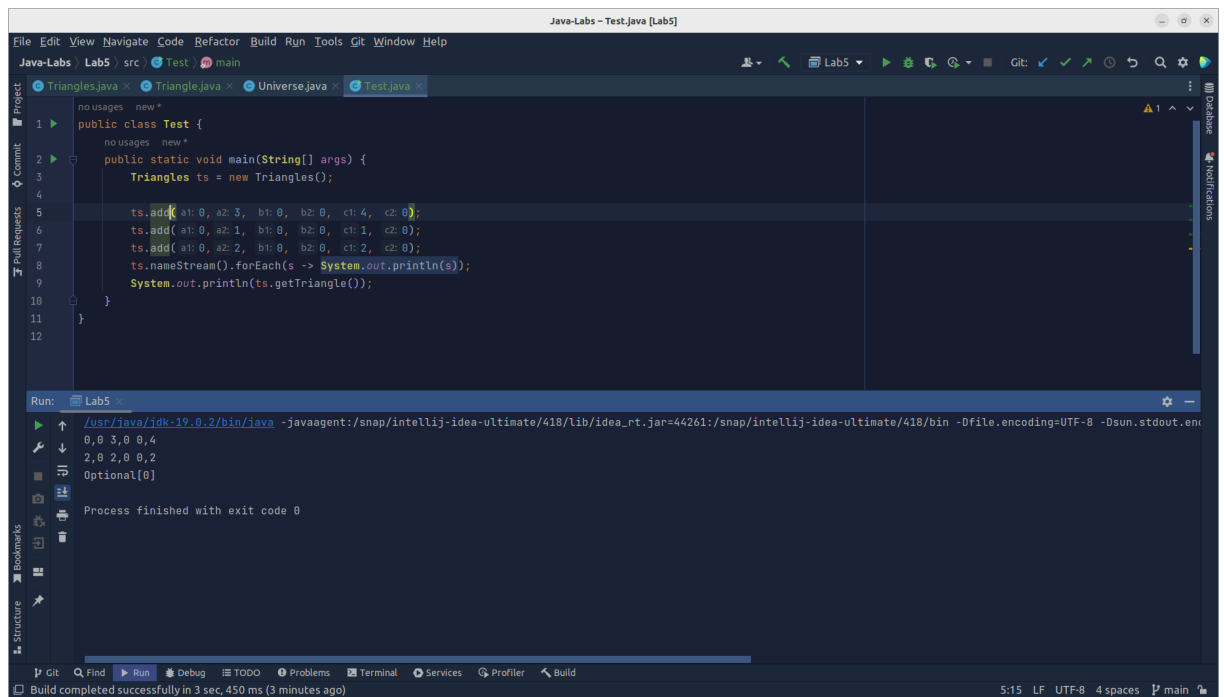


Рис. 1 — Скриншот работы программы

Далее добавим ещё один треугольник

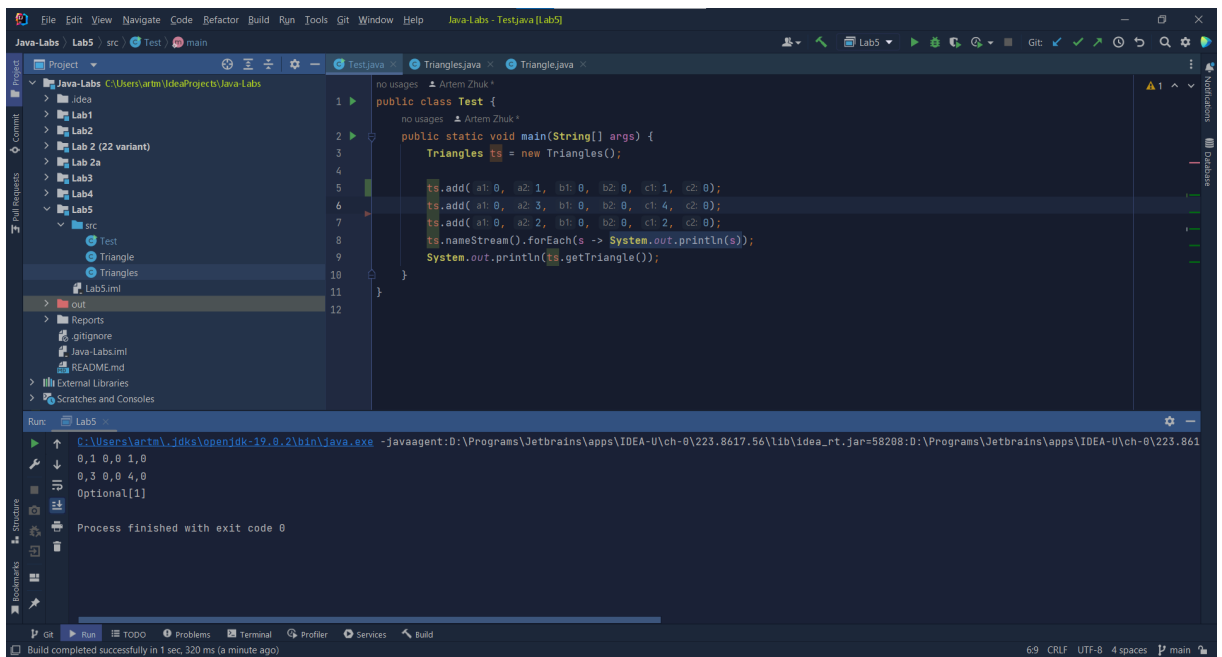


Рис. 2 — Скриншот перерасчёта

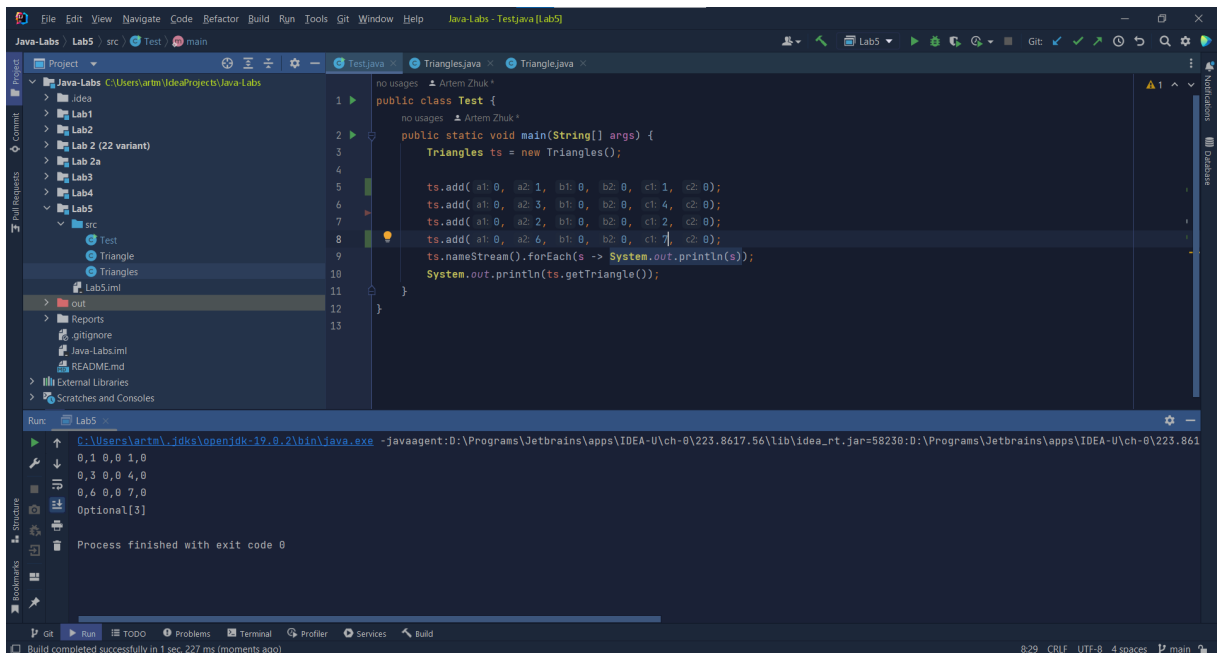


Рис. 3 — Скриншот с 4мя треугольниками

## Листинг 1 — Код класса Triangles

```

1 import java.util.ArrayList;
2 import java.util.HashMap;
3 import java.util.Optional;
4 import java.util.concurrent.atomic.AtomicReference;
5 import java.util.stream.Stream;
6
7 public class Triangles {
8     HashMap<Integer, Triangle> Table;
9     int total;
10    Triangles(){
11        Table = new HashMap<>();
12        total = 0;
13    }
14    void add (Triangle t){
15        Table.put(total, t);
16        total ++;
17    }
18    void add(int a1,int a2, int b1, int b2, int c1, int c2){
19        Table.put(total, new Triangle(total, a1,a2,b1, b2,c1, c2));
20        total++;
21    }
22    /* 1.
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50 }

```

## Листинг 2 — Код класса Triangle

```

1 public class Triangle {
2     int a1, a2, b1, b2, c1, c2, id;
3     double area;
4     double gip;
5     public Triangle(int a1, int a2, int b1, int b2, int c1, int c2, int
id) {
6         this.a1 = a1;
7         this.a2 = a2;
8         this.b1 = b1;
9         this.b2 = b2;
10        this.c1 = c1;
11        this.c2 = c2;
12        this.id = id;
13        this.area = this.area();
14        this.gip = this.getGip();
15    }
16
17
18    public boolean isSquared() {
19        double alen, blen, clen;
20        alen = Math.pow((double) (this.a1 - this.b1), 2) + Math.pow((
double) (this.a2 - this.b2), 2);
21        blen = Math.pow((double) (this.b1 - this.c1), 2) + Math.pow((
double) (this.b2 - this.c2), 2);
22        clen = Math.pow((double) (this.c1 - this.a1), 2) + Math.pow((
double) (this.c2 - this.a2), 2);
23        if (alen + clen == blen) {
24            return true;
25        } else {
26            return false;
27        }
28    }
29    public double getGip(){
30        return Math.pow((double) (this.c1 - this.a1), 2) + Math.pow((
double) (this.c2 - this.a2), 2);
31    }
32    public String toString(){
33        StringBuilder result = new StringBuilder();
34        result.append(this.a1 + "," + this.a2 + " " + this.b1 + "," +
this.b2
35        + " " + this.c1 + "," + this.c2);
36        return result.toString();
37    }
38    public double area(){
39        double alen, blen, clen;
40        alen = Math.pow((double) (this.a1 - this.b1), 2) + Math.pow((
double) (this.a2 - this.b2), 2);
41        blen = Math.pow((double) (this.b1 - this.c1), 2) + Math.pow((
double) (this.b2 - this.c2), 2);
42        clen = Math.pow((double) (this.c1 - this.a1), 2) + Math.pow((
double) (this.c2 - this.a2), 2);
43        alen = Math.sqrt(alen);
44        blen = Math.sqrt(blen);
45        clen = Math.sqrt(clen);
46        double perim = alen + blen + clen;
47        perim /= 2;
48        double area = Math.sqrt(perim *(perim - alen) * (perim -blen) *
(perim - clen));
49        return area;
50    }
51 }

```

### Листинг 3 — Код класса Test

```
1 public class Test {  
2     public static void main(String[] args) {  
3         Triangles ts = new Triangles();  
4  
5         ts.add(0,3, 0, 0, 4, 0);  
6         ts.add(0,1, 0, 0, 1, 0);  
7         ts.add(0,2, 0, 0, 2, 0);  
8         ts.nameStream().forEach(s -> System.out.println(s));  
9         System.out.println(ts.getTriangle());  
10    }  
11 }
```