

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 №5
Монады в языке Java
Вариант №3

Выполнил
студент группы ИУ9-21Б
Лисов Алексей

Москва, 2023

1 Условие

Приобретение навыков использования монад `Optional` и `Stream` в программах на языке. В варианте номер 3 необходимо было реализовать Последовательность отрезков прямых на плоскости, заданных целочисленными координатами двух точек, с операциями: порождение потока, содержащего отрезки из последовательности, каждый из которых параллелен предыдущему отрезку; поиск такого отрезка, что на нём лежат по одному концу каждого из остальных отрезков последовательности. Условие задачи, исходный код и пример работы программы необходимо прислать в формате `LaTeX`.

2 Код решения

Файл `Main.java`

```
import java.util.stream.Stream;

public class Main {
    public static void main(String[] args) {
        SegmentsTable a = new SegmentsTable();
        //a.add(new Point(1, 1), new Point(3, 1));
        //a.add(new Point(1, 2), new Point(3, 2));
        //a.add(new Point(1, 1), new Point(1, 3));
        //a.nameStream();

        a.add(new Point(5, 1), new Point(4, 1));
        a.add(new Point(1, 1), new Point(1, 3));
        a.add(new Point(2, 1), new Point(2, 3));

        System.out.println(a.getProduct());
    }
}
```

Файл `Segment.java`

```
public class Segment {
    public Point fst, scd;
    Segment(Point fst, Point scd) {
        this.fst = fst;
        this.scd = scd;
    }
}
```

```

boolean checkPar(Segment seg2) {
    return (((this.scd.y - this.fst.y) *
        (seg2.scd.x - seg2.fst.x)) ==
        ((this.scd.x - this.fst.x) *
        (seg2.scd.y - seg2.fst.y)));
}

boolean contains(Point a) {
    if ((a.x >= this.fst.x && a.x <= this.scd.x) ||
        (a.x >= this.scd.x && a.x <= this.fst.x)) {
        if ((a.y >= this.fst.y && a.y <= this.scd.y)
            || (a.y >= this.scd.y && a.y <= this.fst.y)) {
            return true;
        }
    }
    return false;
}
}

```

Файл SegmentsTable.java

```

import java.util.*;
import java.util.stream.Stream;

public class SegmentsTable {
    ArrayList<Segment> segments;
    SegmentsTable() {
        segments = new ArrayList<Segment>();
    }

    void add(Segment p) {
        segments.add(p);
        //total += p.count;
    }

    void add(Point fst, Point scd) {
        segments.add(new Segment(fst, scd));
    }

    public void nameStream() {

```

```

        Stream.iterate(1, n -> n + 1)
            .limit(segments.size()-1)
            .filter(x -> (segments.get(x).checkPar(segments.get(x-1))))
            .forEach(x -> System.out.println(x));
    }

    //поиск такого отрезка, что на нём лежат по
    //одному концу каждого из остальных отрезков
    //последовательности.
    public Optional<Segment> getProduct() {
        Optional<Segment> result = segments.stream()
            .filter(segment -> {
                // check if
                //the segment contains one end of
                //each of the remaining segments
                List<Segment> remainingSegments = new ArrayList<>(segments);
                remainingSegments.remove(segment);

                ArrayList<Integer> smth = new ArrayList<>();
                Stream.iterate(0, n -> n+1).limit(remainingSegments.size()-1)
                    .filter(
                        x -> (segment.contains(remainingSegments.get(x).fst)

||

                        segment.contains(remainingSegments.get(x).scd)))
                    .forEach(x -> smth.add(x));

                return (remainingSegments.size() == smth.size());
            })
            .findFirst();

        return result;
    }
}

```

Файл Point.java

```

public class Point {
    int x, y;
}

```

```

    public Point(int x, int y) {
        this.x = x;
        this.y = y;
    }
}

```

3 Скриншоты

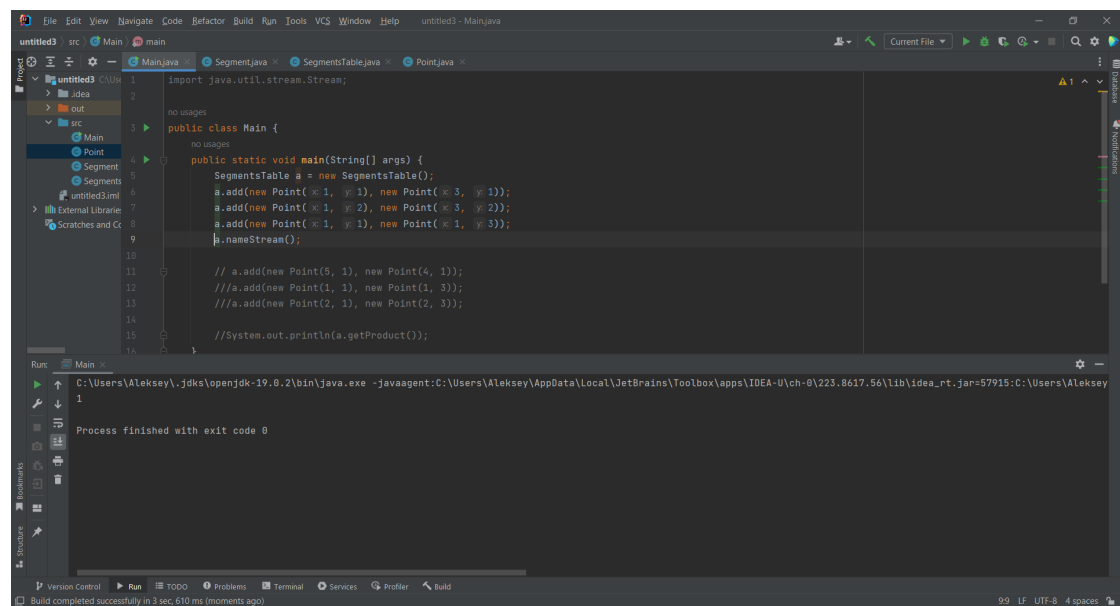


Рис. 1: Вывод программы

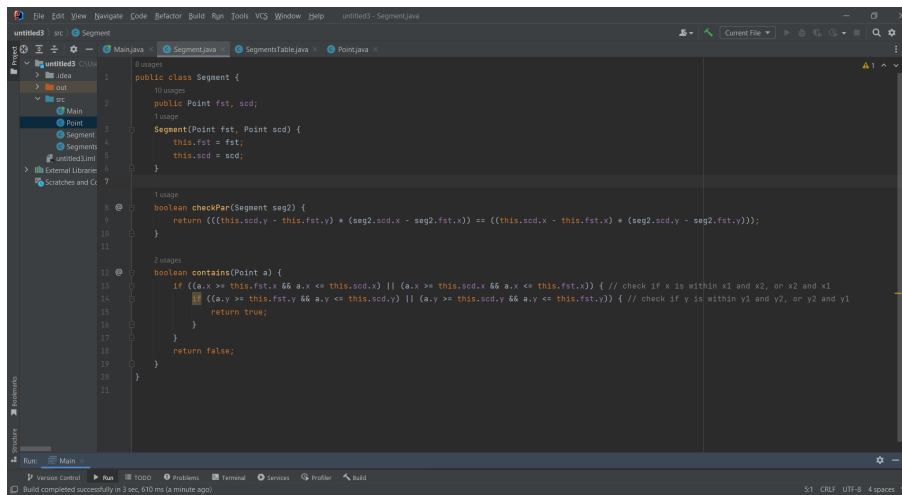


Рис. 2: Файл Segment.java

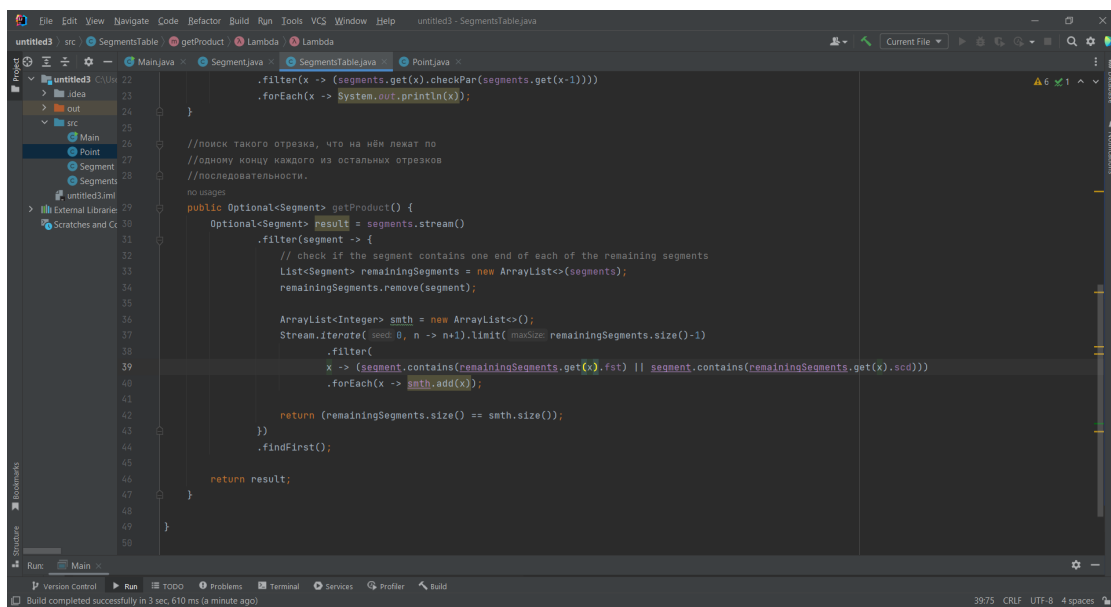


Рис. 3: Файл SegmentsTable.java