



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«МИРЭА – Российский технологический университет»

РТУ МИРЭА

**Институт информационных технологий (ИТ)
Кафедра инструментального и прикладного программного обеспечения (ИиППО)**

Дисциплина «Программирование на языке Джава»

**ОТЧЕТ
ПО ПРАКТИЧЕСКОМУ ЗАНЯТИЮ №27-28**

Выполнил студент группы ИНБО-02-20

Лукияненко Д.В.

Принял

Степанов П.В.

Практическая работа выполнена «__» _____ 2021 г.

«_____»
Отметка о выполнении

«__» _____ 2021 г.

Москва – 2021 г.

СОДЕРЖАНИЕ

Цель работы	3
Задание	3
Репозиторий	3
Выполнение работы	3
Код выполненной работы	4
Тестирование программы	4
Вывод	8

Цель работы

Цель данной практической работы – Научиться уменьшать изображения.

Задание

Напишите программу, которая уменьшает изображения на основе кода который приведен в данном пакете задания. Она должна запускать число потоков, равное количеству ядер процессора вашего компьютера.

Обязательно запустите исходный код проекта и осознайте что однопоточна обработка изображений медленная. Ваша основная задача сделать обработку большого количества изображений быстрее за счет многопоточности, но без вылета ошибок невозможности создать поток и не перегружая слишком сильно систему юзера.

Репозиторий

Ссылка: <https://github.com/neluckoff/mirea-java-lessons/tree/master/src/ru/luckoff/mirea/exercies27and28>

Выполнение работы

Для выполнения данного практического задания нам уже был дан код программы, в который нам нужно было лишь добавить количество потоков и работу с ними для уменьшения размера изображений.

Также в итоговом коде было изменено место сохранения и место выбора изображений. Это связано с тем, что все практические работы находятся в одном проекте и папки, принадлежащие определенной работе, теряться не должны.

Код выполненной работы

```
public class Main implements Runnable {  
    public static void main(String[] args) {  
        Main image = new Main();  
        int cores = Runtime.getRuntime().availableProcessors();  
        ArrayList<Thread> threads = new ArrayList<>();  
        for(int i = 0; i < cores; i++) {  
            Thread thread = new Thread(image);  
            thread.setName(String.valueOf(i));  
            threads.add(thread);  
        }  
        for (Thread thread : threads){  
            thread.start();  
        }  
    }  
}
```

Рисунок 1 – Запуск программы

```
@Override  
public void run() {  
    resize(Integer.parseInt(Thread.currentThread().getName()));  
}
```

Рисунок 2 – Метод run()

```

private void resize(int i) {
    int cores = Runtime.getRuntime().availableProcessors();
    String srcFolder = "src/ru/luckoff/mirea/exercies27and28/images";
    String dstFolder = "src/ru/luckoff/mirea/exercies27and28/new_images";

    File srcDir = new File(srcFolder);

    long start = System.currentTimeMillis();

    File[] files = srcDir.listFiles();
    assert files != null;
    cores=files.length/cores;
    try {
        if (!Files.exists(Paths.get(dstFolder))) {
            Files.createDirectories(Paths.get(dstFolder));
        }
        for (int l=i;l<cores*(i+1);l++) {
            File file = files[l];
            BufferedImage image = ImageIO.read(file);
            if (image == null) {
                continue;
            }

            int newWidth = image.getWidth() / 4;
            int newHeight = (int) Math.round(
                image.getHeight() / (image.getWidth() / (double) newWidth)
            );
            BufferedImage newImage = new BufferedImage(
                newWidth, newHeight, BufferedImage.TYPE_INT_RGB
            );

            int widthStep = image.getWidth() / newWidth;
            int heightStep = image.getHeight() / newHeight;

```

Рисунок 3 – Метод resize (1)

```

        for (int x = 0; x < newWidth; x++) {
            for (int y = 0; y < newHeight; y++) {
                int rgb = image.getRGB(x * widthStep, y * heightStep);
                newImage.setRGB(x, y, rgb);
            }
        }

        File newFile = new File(pathname: dstFolder + "/" + file.getName());
        ImageIO.write(newImage, formatName: "jpg", newFile);
    }
} catch (Exception ex) {
    ex.printStackTrace();
}

System.out.println("Время работы: " + (System.currentTimeMillis() - start));
}
}

```

Рисунок 4 – Метод resize (2)

Тестирование программы

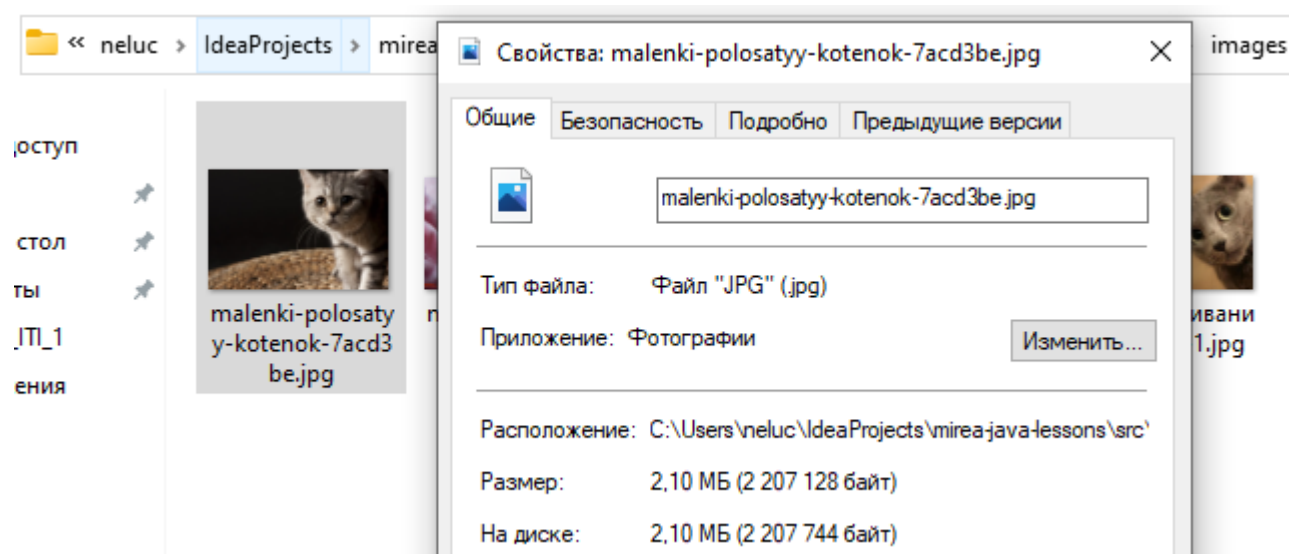


Рисунок 5 – Картинка до обработки программой

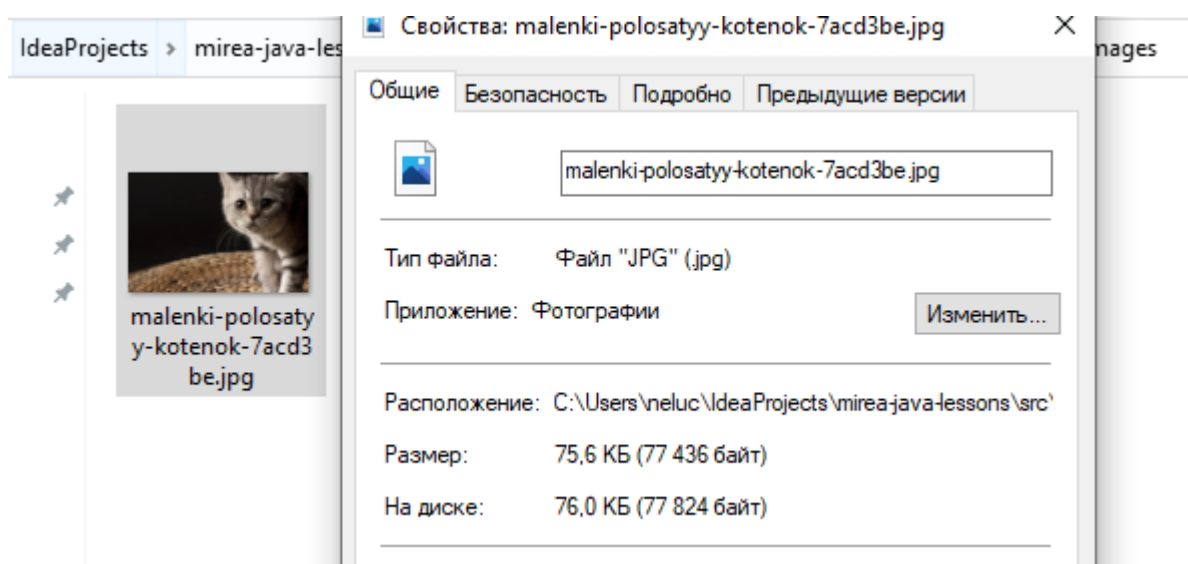


Рисунок 6 – Картинка после обработки программой

Вывод

В результате выполнения данной практической работы я научился уменьшать изображения при помощи языка программирования Java.