



Міністерство освіти і науки України
Національний технічний університет України
“Київський політехнічний інститут імені Ігоря Сікорського”
Факультет інформатики та обчислювальної техніки
Кафедра інформаційних систем та технологій

Лабораторна робота № 4

Потоки вводу-виводу

Виконали
студенти групи ІА-23:
Семашко Олександр,
Ширяєв Даніїл,
Степанов Нікіта.

Перевірив:
Колеснік В. М.

Хід роботи:

1. Ознайомитись з API класів та інтерфейсів для здійснення операцій вводу-виводу. Особливу увагу звернути на такі класи та інтерфейси:

- InputStream
 - FileInputStream
- OutputStream
 - FileOutputStream
- Reader
 - FileReader
- Writer
 - FileWriter
- AutoCloseable
 - Closable
- IOException

2. Виконати завдання з таблиці 2 відповідно до свого варіанту у таблиці 1.

Завдання № 1

| | |
|---|---|
| 3 | <code>void copyNoEmptyStrings(String source, String destination)</code> Скопіювати текстовий файл <i><source></i> у <i><destination></i> , видаливши в процесі копіювання усі порожні рядки. |
|---|---|

Завдання № 2

| | |
|---|--|
| 9 | <code>long longestZeroSequenceLength(String filename)</code> Знайти довжину найдовшої послідовності двійкового файлу, що складається з байтів зі значенням 0. |
|---|--|

```

no usages
1 ▶ public class Main {
    no usages
2 ▶     public static void main(String[] args) {
3         Task1.copyNoEmptyStrings( source: "input/test.txt", destination: "output/output.txt");
4         System.out.println("Longest zero sequence: " +
5             Task2.longestZeroSequenceLength( filename: "out/production/untitled/Main.class"));
6     }
7 }

```

```

1 import java.io.*;
2
3 public class Task1 {
    1 usage
4     static void copyNoEmptyStrings(String source, String destination) {
5         try (BufferedReader in = new BufferedReader(new FileReader(source))) {
6             try (FileWriter writer = new FileWriter(destination)) {
7                 String line;
8                 while((line = in.readLine()) != null) {
9                     if (!line.isEmpty()) {
10                         writer.write( str: line + "\n");
11                     }
12                 }
13             } catch (IOException e) {
14                 e.printStackTrace();
15             }
16         } catch (IOException e) {
17             e.printStackTrace();
18         }
19     }
20 }
21

```

```

1 > import ...
3
    1 usage
4 public class Task2 {
    1 usage
5     static long longestZeroSequenceLength(String filename) {
6         long longestZeroSequenceLength = 0;
7         long currantZeroSequenceLength = 0;
8         try (FileInputStream stream = new FileInputStream(filename)) {
9             byte[] byteList = stream.readAllBytes();
10            for (byte bt: byteList) {
11                if (bt == 0) {
12                    currantZeroSequenceLength++;
13                } else {
14                    if (longestZeroSequenceLength < currantZeroSequenceLength) {
15                        longestZeroSequenceLength = currantZeroSequenceLength;
16                        currantZeroSequenceLength = 0;
17                    }
18                }
19            }
20            return longestZeroSequenceLength;
21        } catch (IOException e) {
22            e.printStackTrace();
23            return -1;
24        }
25    }
26 }
27
28

```

Longest zero sequence: 25

Висновок: на цій лабораторній роботі ми вивчили та навчилися працювати з потоками вводу-виводу, зрозуміли різницю між InputStream і Reader, вивчили

конструкцію try-with-resource, познайомилися з методами flush() та close(), покращили навички кодування на java.