



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 КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)

НАПРАВЛЕНИЕ ПОДГОТОВКИ 09.04.01 Информатика и вычислительная техника

МАГИСТЕРСКАЯ ПРОГРАММА 09.04.01/07 Интеллектуальные системы анализа,
обработки и интерпретации больших данных.

О Т Ч Е Т

по лабораторной работе № 7

Вариант № 17

Название: Строки и регулярные выражения

Дисциплина: Языки программирования для работы с большими данными

Студент

ИУ6-23М

(Группа)

(Подпись, дата)

М.О. Усманов

(И.О. Фамилия)

Преподаватель

(Подпись, дата)

П.В. Степанов

(И.О. Фамилия)

Москва, 2022

Цель работы

Получение первичных навыков работы со строками и регулярными выражениями языка программирования Java.

Ход работы

Задание 1.

– В зависимости от признака (0 или 1) в каждой строке текста удалить указанный символ везде, где он встречается, или вставить его после k-го символа.

– Из небольшого текста удалить все символы, кроме пробелов, не являющиеся буквами. Между последовательностями подряд идущих букв оставить хотя бы один пробел.

Листинг 1 – Код программы первого задания

```
public class App {
    public static void main(String[] args) throws Exception {
        String test = "Hello, World!";
        System.out.println(SymbolOps(test, 'l', 0, 0));
        System.out.println(SymbolOps(test, 'l', 1, 2));

        String line = "Helloworld11@ whiltespalces$ owari";
        System.out.println(PurgeNonLetters(line));
    }

    public static String SymbolOps(String input, Character symbol, int
sign, int k) {
        if (sign == 0) {
            return input.replaceAll(symbol.toString(), "");
        } else if (sign == 1) {
            return input.substring(0, k) + symbol + input.substring(k);
        }
        return "";
    }

    public static String PurgeNonLetters(String input) {
        return input.replaceAll("[^a-zA-Z ]", "");
    }
}
```

Приведем результаты выполнения данного кода.

```
PS E:\Projects\Java\Repo\lab_7_var_1_7_8> & 'C:\Program
objects\Java\Repo\lab_7_var_1_7_8\bin' 'App'
Heo, Word!
Hello, World!
Helloword whitespaces owari
PS E:\Projects\Java\Repo\lab_7_var_1_7_8>
```

Рисунок 1 – Результат выполнения варианта задания 1

Задание 2.

- В тексте найти и напечатать все слова максимальной и все слова минимальной длины.
- Напечатать квитанцию об оплате телеграммы, если стоимость одного слова задана.

Листинг 2 – Код выполнения задания 2

```
import java.util.ArrayList;

public class App {
    public static void main(String[] args) throws Exception {
        String line = "Lorem ipsum dolor sit amet, consectetur adipiscing
mit elit Phasellus vitae";
        PrintMaxMin(line);

        String telegram = "Ichi ni    san yon go roku    shichi";
        System.out.println(CalculateCost(telegram, 10));
    }

    public static int CalculateCost(String input, int price) {
        int cost = 0;

        String[] str = input.split(" ");
        cost = str.length * price;

        return cost;
    }

    public static void PrintMaxMin(String input) {
        String[] max = maxWords(input);
        String[] min = minWords(input);

        for (String w : max) {
            System.out.println(w);
        }

        for (String w : min) {
            System.out.println(w);
        }
    }
}
```

```

    }
}

private static String[] maxWords(String input) {
    String[] str = input.split(" ");
    if (str.length==0) return null;

    ArrayList<String> longest = new ArrayList<>();
    int longest_length = 0;
    for (String word : str){

        if (word.length() > longest_length) {
            longest.clear();
            longest_length = word.length();
            longest.add(word);
        } else if (word.length() == longest_length) {
            longest.add(word);
        }
    }

    String[] result = new String[longest.size()];
    result = longest.toArray(result);
    return result;
}

private static String[] minWords(String input) {
    String[] str = input.split(" ");
    if (str.length==0) return null;

    ArrayList<String> least = new ArrayList<>();
    int least_length = Integer.MAX_VALUE;
    for (String word : str){

        if (word.length() < least_length) {
            least.clear();
            least_length = word.length();
            least.add(word);
        } else if (word.length() == least_length) {
            least.add(word);
        }
    }

    String[] result = new String[least.size()];
    result = least.toArray(result);
    return result;
}
}

```

Результаты выполнения задания приведем на рисунке далее.

```
PS E:\Projects\Java\Repo\lab_7_var_2_7_8> & 'C:\Program Files\Amazon Corretto\jdk17.0.2_8\bin\j
objects\Java\Repo\lab_7_var_2_7_8\bin' 'App'
consectetur
sit
mit
70
PS E:\Projects\Java\Repo\lab_7_var_2_7_8>
```

Рисунок 2 – Выполненный код задания 2

Задание 3.

– Текст шифруется по следующему правилу: из исходного текста выбирается 1, 4, 7, 10-й и т.д. (до конца текста) символы, затем 2, 5, 8, 11-й и т.д. (до конца текста) символы, затем 3, 6, 9, 12-й и т.д. Зашифровать заданный текст.

– На основании правила кодирования, описанного в предыдущем примере, расшифровать заданный набор символов.

Листинг 3 – Код основной программы

```
public class App {
    public static void main(String[] args) throws Exception {
        String test = "abcdefgh";
        String coded = EncodeString(test);
        String decoded = EncodeString(coded);
        System.out.println(coded);
        System.out.println(decoded);
    }

    public static String EncodeString(String input) {
        int len = input.length();
        String result = "";

        int offset = 0;
        while (result.length() < len) {
            int index = 0 + offset;

            while (index < input.length()) {
                result += input.charAt(index);
                index += 3;
            }

            offset++;
        }

        return result;
    }
}
```

Приведем результат выполнения программы:

```
PS E:\Projects\Java\Repo\lab_7_var_3_7_8>
objects\Java\Repo\lab_7_var_3_7_8\bin' 'App'
adgbehcf
abcdefgh
PS E:\Projects\Java\Repo\lab_7_var_3_7_8>
```

Рисунок 3 – Зашифрованный и расшифрованный текст

Задание 4.

- Подсчитать, сколько слов в заданном тексте начинается с прописной буквы.
- Подсчитать, сколько раз заданное слово входит в текст.

Листинг 4 – Код программы

```
public class App {
    public static void main(String[] args) throws Exception {
        String input = "Lorem ipsum dolor sit amet Consectetur adipiscing
elit Praesent blandit amet";
        String[] str = input.split(" ");

        int capitaled = 0;
        for (String w : str) {
            if (w.length() > 0) {
                if (Character.isUpperCase(w.charAt(0)) == true)
                    capitaled++;
            }
        }
        System.out.println(capitaled);

        String word = "amet";
        int count = 0;
        for (String w : str) {
            if (w.toLowerCase().equals(word.toLowerCase())) count++;
        }
        System.out.println(count);
    }
}
```

```
PS E:\Projects\Java\Repo\lab
objects\Java\Repo\lab_7_var_4
3
2
PS E:\Projects\Java\Repo\lab
```

Рисунок 4 – Результат выполнения работы

Местоположение репозитория с файлами проекта

Файлы проекта расположены в репозитории веб-платформы для совместной разработки Github. Местоположение в репозитории:

https://github.com/s314/big-data-studies/tree/main/lab_7_var_1_7_8

https://github.com/s314/big-data-studies/tree/main/lab_7_var_2_7_8

https://github.com/s314/big-data-studies/tree/main/lab_7_var_3_7_8

https://github.com/s314/big-data-studies/tree/main/lab_7_var_4_7_8

Вывод

По итогам выполнения лабораторной работы были получены навыки программирования с использованием строк в языке Java. Также были изучены способы обработки строк при помощи регулярных выражений.