



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 14**

**Название:** Строки и регулярные выражения

**Дисциплина:** Языки программирования для работы с большими  
данными

Студент

ИУ6-23М

(Группа)

\_\_\_\_\_  
(Подпись, дата)

В.Е. Санталов

(И.О. Фамилия)

Преподаватель

\_\_\_\_\_  
(Подпись, дата)

П.В. Степанов

(И.О. Фамилия)

Москва, 2022

**Цель работы:** получение навыков работы со строками и регулярными выражениями в Java.

### Задание 1:

4. В тексте слова заданной длины заменить указанной подстрокой, длина которой может не совпадать с длиной слова.
5. В тексте после k-го символа вставить заданную подстроку.

### Подзадача 1.

Код программы:

```
public class Laba1_4 {  
    public static void main(String[] args) throws Exception {  
        String text = "Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s";  
        String textNew = "";  
  
        String subWord = args[1];  
  
        for (String word : text.split(regex: " ")) textNew += word.matches("[a-zA-Za-яА-Я]{ " + args[0] + "}") ? subWord + " " : word + " ";  
  
        System.out.println(textNew);  
    }  
}
```

### Результат выполнения программы:

```
Desktop\Java\lr7\Woe\1\bin' 'Labai_4' 5 shit  
shit shit is simply shit text of the printing and typesetting industry. shit shit has been the industry's standard shit text ever shit the 1500s, when an unknown printer took a  
galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unc  
hanged. It was popularised in the 1960s with the release of Letraset sheets containing shit shit passages, and more recently with desktop publishing software like shit PageMake  
r including versions of shit Ipsum.
```

### Подзадача 2.

Код программы:

```
public class Laba1_5 {  
    public static void main(String[] args) throws Exception {  
        String text = "Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s";  
        String textNew = "";  
  
        int k = Integer.parseInt(args[0]);  
        String substring = args[1];  
  
        textNew += text.substring(beginIndex: 0, k) + substring + text.substring(k);  
  
        System.out.println(textNew);  
    }  
}
```

### Результат выполнения программы:

```
tailsInExceptionMessages' '-cp' 'C:\Users\time2\Desktop\Java\lr7\Woe\1\bin' 'Labai_5' 9 "i love java "  
Lorem Ipsi love java um is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unk  
nown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remain  
ing essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing softwa  
re like Aldus PageMaker including versions of Lorem Ipsum.
```

## Задание 2:

4. Найти, каких букв, гласных или согласных, больше в каждом предложении текста.

5. В стихотворении найти количество слов, начинающихся и заканчивающихся гласной буквой.

## Подзадача 1.

Код программы:

```
import java.util.regex.Matcher;
import java.util.regex.Pattern;

public class Laba2_4 {
    Run | Debug
    public static void main(String[] args) {
        String text = "Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s,

        Pattern vowel = Pattern.compile(regex: "[aeiouy]");
        Pattern consonant = Pattern.compile(regex: "abcfgghjklmnpqrstvwxyz");

        for (String sent : text.split(regex: "[.!?]")) {
            int vowelCount = 0;
            int consonantCount = 0;

            Matcher vMatcher = vowel.matcher(sent);
            Matcher cMatcher = consonant.matcher(sent);

            while (vMatcher.find()) {
                vowelCount++;
            }

            while (cMatcher.find()) {
                consonantCount++;
            }

            System.out.print("В предложении '" + sent + "' ");
            if (vowelCount > consonantCount) System.out.println(x: "больше всего гласных букв.");
            else if (vowelCount < consonantCount) System.out.println(x: "больше всего согласных букв.");
            else System.out.println(x: "гласных и согласных букв поровну.");
        }
    }
}
```

## Результат работы программы:

```
В предложении 'Lorem Ipsum is simply dummy text of the printing and typesetting industry' больше всего гласных букв.
В предложении 'Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book' больше всего гласных букв.
В предложении 'It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged' больше всего гласных букв.
В предложении 'It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum' больше всего гласных букв.
```

## Подзадача 2.

Код программы:



### Задание 3:

4. Во всех вопросительных предложениях текста найти и напечатать без повторений слова заданной длины.
5. В каждом предложении текста поменять местами первое слово с последним, не изменяя длины предложения.

### Подзадача 1.

Код программы:

```
import java.util.ArrayList;
import java.util.regex.Matcher;
import java.util.regex.Pattern;

public class Laba3_4 {
    Run | Debug
    public static void main(String[] args) {
        String text = "Базаров был великий охотник до женщин и до женской красоты, но любовь в смысле идеальном, или, как он выражался, романтическом, называл белибердой, непр";
        String patternStr = "[^.?!]*[?]*";

        Pattern p = Pattern.compile(patternStr);
        Matcher m = p.matcher(text);

        ArrayList<String> words = new ArrayList<String>();

        while (m.find()) {
            String sent = m.group();
            System.out.println(sent);

            for (String word : sent.split(regex: " ")) {
                word = word.replaceAll(regex: "[.?.]", replacement: "");
                if (!words.contains(word)) words.add(word);
            }
        }

        System.out.println(words);
    }
}
```

### Результат работы программы:

```
Базаров был великий охотник до женщин и до женской красоты, но любовь в смысле идеальном, или, как он выражался, романтическом, называл белибердой, непростительную дурию, считал рыцарские чувства чем-то вроде уродства
или болезни и не однажды выражал свое удивление: почему не посадили в желтый дом Тоггенбурга со всеми миннезингерами и трубадурами?
[Базаров, был, великий, охотник, до, женщин, и, женской, красоты, но, любовь, в, смысле, идеальном, или, как, он, выражался, романтическом, называл, белибердой, непростительную, дурию, считал, рыцарские, чувства, чем-то, вроде, уродства, болезни, не, однажды, выражал, свое, удивление:, почему, посадили, желтый, дом, Тоггенбурга, со, всеми, миннезингерами, трубадурами]
```

### Подзадача 2.

Код программы:

```

import java.util.regex.Matcher;
import java.util.regex.Pattern;

public class Laba3_5 {
    Run | Debug
    public static void main(String[] args) {
        String text = "Вступив в рабочее движение в 70-х годах XIX века, Каутский вначале разделял взгляды, далёкие от марксизма. Затем он познакомился с Марксом и Энгельсом и

        Matcher sentenceM = Pattern.compile(regex: "[^.?!](.)*[.?!]").matcher(text);
        Pattern fWordP = Pattern.compile(regex: "[A-Я]+");
        Pattern lWordP = Pattern.compile(regex: "[A-Я]+.$");

        while (sentenceM.find()) {
            String sentence = sentenceM.group().toLowerCase();
            if (sentence.startsWith(prefix: " ")) sentence = sentence.substring(beginIndex: 1);

            Matcher fWordM = fWordP.matcher(sentence);
            Matcher lWordM = lWordP.matcher(sentence);
            fWordM.find();
            lWordM.find();

            String firstWord = fWordM.group();

            String lastWord = lWordM.group();
            lastWord = lastWord.substring(beginIndex: 0, lastWord.length() - 1);

            sentence = sentence.replaceFirst(lastWord, firstWord);

            lastWord = lastWord.substring(beginIndex: 0, endIndex: 1).toUpperCase() + lastWord.substring(beginIndex: 1);
            sentence = sentence.replaceFirst(firstWord, lastWord);

            System.out.println(sentence);
        }
    }
}

```

## Результат работы программы:

Марксизма в рабочее движение в 70-х годах XIX века, каутский вначале разделял взгляды, далёкие от вступив. Непосредственным он познакомился с марксом и энгельсом и стал сторонником их учения, хотя весьма затем. Филистерству и энгельс резко критиковали ошибки каутского, его склонность к маркс.

'Каутского учение карла маркса' - едва ли не самое известное из всего, что вышло из-под пера экономическое.

Рукописи в 1886 г., эта книга была, по свидетельству каутского, просмотрена энгельсом в написанная.

Подлинника этой работы каутского в том, что автор даёт здесь популярное изложение ?капитала? маркса (главным образом в пределах первого тома), близко придерживаясь особенности.

Маркса широко цитирует он.

Текст и там, где каутский передаёт содержание 'капитала', не прибегая к цитатам, он старается воспроизводить ход мысли маркса, используя его аргументацию и даже художественные образы, оживляющие но.

## Задание 4:

- В тексте исключить подстроку максимальной длины, начинающуюся и заканчивающуюся заданными символами.
- Заменить все одинаковые рядом стоящие символы в тексте одним символом.

## Подзадача 1.

Код программы:

```

import java.util.regex.Matcher;
import java.util.regex.Pattern;

public class Laba4_4 {
    Run | Debug
    public static void main(String[] args) {
        String text = "aaaaazzzzzzaaazzzaammaaaaazmzzzzmzaaaaa";

        Matcher matcher = Pattern.compile(args[0] + ".*" + args[1]).matcher(text);
        matcher.find();

        System.out.println("Text before: " + text);

        text = text.replace(matcher.group(), replacement: "");
        System.out.println("Text after: " + text);
    }
}

```

**Результат работы программы:**

```

\time2\Desktop\Java\clone\lr7\bin' 'Laba4_4' m m
Text before: aaaaazzzzzzaaazzzaammaaaaazmzzzzmzaaaaa
Text after: aaaaazzzzzzaaazzzaazaaaaa
PS C:\Users\time2\Desktop\Java\clone\lr7>

```

**Подзадача 2.**

Код программы:

