



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 18**

**Название:** Работа со строками и регулярными выражениями

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

Студент

ИУ6-23М

(Группа)

\_\_\_\_\_  
(Подпись, дата)

Д.Н. Хныкин

(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.

## Выполнение

### Задание 1:

1. Из небольшого текста удалить все символы, кроме пробелов, не являющиеся буквами. Между последовательностями подряд идущих букв оставить хотя бы один пробел.
2. Из текста удалить все слова заданной длины, начинающиеся на согласную букву.

Листинг выполнения подзадачи 1

#### #n81.java

```
package com.aranei.var1;

public class n81 {
    public void nv81() {
        String line = "1234НННН89__++hjkjhj    jkg    132    pp";
        System.out.println(line);
        System.out.println(deleteLetters(line));
    }
    public static String deleteLetters(String input) {
        return input.replaceAll("[^a-zA-Z ]", " ").replaceAll("\\s+", " ").trim();
    }
}
```

#### #n91.java

```
package com.aranei.var1;

import java.util.Arrays;

public class n91 {
    public void nv91() {
        String text = "Привет как дела где телеграмма";
        String[] symbols = "б в г д ж з й к л м н п р с т ф х ц ч ш щ".split(" ");
        int number = 5;
        String[] textArray = text.split(" ");
        System.out.println(text);
        for (int i = 0; i != textArray.length; i++) {
            String symbol = String.valueOf(textArray[i].charAt(0)).toLowerCase();
            if (textArray[i].length() == number && Arrays.asList(symbols).contains(symbol)) {
                text = text.replace(textArray[i], "");
            }
        }
        System.out.println(text.replaceAll("\\s+", " ").trim());
    }
}
```

### Задание 2:

3. Напечатать квитанцию об оплате телеграммы, если стоимость одного слова задана.

#### 4. В стихотворении найти одинаковые буквы, которые встречаются во всех словах.

##### Листинг выполнения подзадачи 2

###### #n82.java

```
package com.aranei.var2;

public class n82 {
    public void nv82(){
        String telegram = "Привет как дела где телеграмма";
        int cost = 0;
        int price = 15;
        String[] str = telegram.split(" ");
        cost = str.length * price;
    }
}
```

###### #n92.java

```
package com.aranei.var2;

public class n92 {
    public void nv92(){
        String text = "Привет каки делаи игде телеграммаи";
        String[] words = text.split(" ");
        boolean isContain;
        for (int i = 0; i < words[0].length(); i++) {
            isContain = true;
            for (int j = 1; j < words.length; j++) {
                isContain &= words[j].contains(String.valueOf(words[0].charAt(i)));
            }
            if (isContain) System.out.print(words[0].charAt(i) + " ");
        }
    }
}
```

#### Задание 3:

1. На основании правила кодирования, описанного в предыдущем примере, расшифровать заданный набор символов.
2. Напечатать слова русского текста в алфавитном порядке по первой букве. Слова, начинающиеся с новой буквы, печатать с красной строки.

##### Листинг выполнения подзадачи 3

###### #n83.java

```
package com.aranei.var3;

public class n83 {
    public void nv83(){
        String test = "123123123";
        String coded = EncodeDecodeString(test);
        String decoded = EncodeDecodeString(coded);
        System.out.println(test);
        System.out.println(coded);
        System.out.println(decoded);
    }
    public static String EncodeDecodeString(String input) {
        int len = input.length();
        StringBuilder result = new StringBuilder();

        int position;
        for(int i=0; i<3; i++){
            position = i;
            while (position < input.length()) {
```

```

        result.append(input.charAt(position));
        position += 3;
    }
}
return result.toString();
}
}

```

#### #n93.java

```

package com.aranei.var3;

import java.util.Arrays;

public class n93 {
    public void nv93() {
        String[] symbols = "а б в г д е ё ж з и й к л м н о п р с т у ф х ц ч ш щ ъ ы ь э ю".split(" ");
        String text = "ш шшш ю а ав ава б бг";
        String[] words = text.split(" ");
        boolean checkN = false;

        for (int i = 0; i < symbols.length; i++) {
            for (int j = 0; j < words.length; j++) {
                if (String.valueOf(words[j].charAt(0)).toLowerCase().equals(symbols[i])) {
                    System.out.print(words[j] + " ");
                    checkN = true;
                }
            }
            if (checkN) System.out.println();
            checkN = false;
        }
    }
}

```

### Задание 4:

3. Подсчитать, сколько раз заданное слово входит в текст.
4. Преобразовать каждое слово в тексте, удалив из него все последующие (предыдущие) вхождения первой (последней) буквы этого слова.

Листинг выполнения подзадачи 4

#### #n84.java

```

package com.aranei.var4;

public class n84 {
    public void nv81(){
        String input = "и и иии ф ф аа аа а";
        String[] str = input.split(" ");

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

```

#### #n94.java

```

package com.aranei.var4;

import java.util.Arrays;

public class n94 {
    public void nv94() {
        String text = "Привет как дела где телеграмма";
        String[] words = text.split(" ");
    }
}

```

```
        System.out.println(Arrays.toString(words));  
        for (int i = 0; i != words.length; i++) {  
            String w1 = String.valueOf(words[i].charAt(0));  
            words[i] = w1 + words[i].replace(w1, "");  
        }  
        System.out.println(Arrays.toString(words));  
    }  
}
```

### **Ссылка на программное решение**

Программное решение представлено в репозитории распределённой системы управления версиями Git:

<https://github.com/Aranei99/labsJava/tree/main/LB7>

### **Вывод**

При выполнении лабораторной работы были получены навыки работы со строками и регулярными выражениями в Java.