



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

**О Т Ч Е Т**

**по лабораторной работе № 7**

**Название:** Строки. Регулярные выражения.

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

Студент

ИУ6-23М

(Группа)

\_\_\_\_\_  
(Подпись, дата)

Д.В. Авдонин

(И.О. Фамилия)

Преподаватель

\_\_\_\_\_  
(Подпись, дата)

\_\_\_\_\_  
(И.О. Фамилия)

Москва, 2022

## Лабораторная работа № 7

### Задание:

2. В русском тексте каждую букву заменить ее порядковым номером в алфавите. При выводе в одной строке печатать текст с двумя пробелами между буквами, в следующей строке внизу под каждой буквой печатать ее номер.

### Ход работы:

#### Код программы:

```
private static final String text = "В русском тексте каждую букву заменить ее порядковым номером в \n" +
"алфавите. При выводе в одной строке печатать текст с двумя\n" +
"пробелами между буквами, в следующей строке внизу\n" +
"под каждой буквой печатать ее номер.";
private static final String alphabet = "абвгдеёжзийклмнопрстуфхцщъыьэюя";

private static void printSymbolLine(String line) {
    for (char symbol : line.toCharArray()) {
        System.out.print(" " + symbol + " ");
    }
    System.out.println();
}

private static void printNumberLine(String line) {
    int number;
    for (char sym : line.toCharArray()) {
        number = alphabet.indexOf(sym);
        if (number == -1){
            number = alphabet.toUpperCase(Locale.ROOT).indexOf(sym);
            if (number == -1){
                System.out.print("-- ");
                continue;
            }
        }
        System.out.printf("%2d ",(number + 1));
    }
    System.out.println();
}

public static void main(String[] args){
    String[] lines = text.split("\\n");
    for (String line : lines) {
        line = line.replaceAll("\\s+", "");
        printSymbolLine(line);
        printNumberLine(line);
    }
}
```

```

В р у с с к о м т е к с т е к а ж д у ю б у к в у з а м е н и т ь е е п о р я д к о в ы м н о м е р о м в
3 18 21 19 19 12 16 14 20 6 12 18 20 6 12 1 8 5 21 32 2 21 12 3 21 9 1 14 6 15 10 20 30 6 6 17 16 18 33 5 12 16 3 29 14 15 16 14 6 18 16 14 3
а л ф а в и т е . П р и в ы в о д е в о д н о й с т р о к е п е ч а т а т ь т е к с т с д в у м я
1 13 22 1 3 10 20 6 -- 17 18 10 3 29 3 16 5 6 3 16 5 15 16 11 19 20 18 16 12 6 17 6 25 1 20 1 20 30 20 6 12 19 20 19 5 3 21 14 33
п р о б е л а м и м е ж д у б у к в а м и , в с л е д у ю щ е й с т р о к е в н и з у
17 18 16 2 6 13 1 14 10 14 6 8 5 21 2 21 12 3 1 14 10 -- 3 19 13 6 5 21 32 27 6 11 19 20 18 16 12 6 3 15 10 9 21
п о д к а ж д о й б у к в о й п е ч а т а т ь е е н о м е р .
17 16 5 12 1 8 5 16 11 2 21 12 3 16 11 17 6 25 1 20 1 20 30 6 6 15 16 14 6 18 --

```

Рисунок 1 – Результат работы программы

### Задание:

3. В тексте после буквы Р, если она не последняя в слове, ошибочно напечатана буква А вместо О. Внести исправления в текст.

### Ход работы:

#### Код программы:

```

public static void main(String[] args){
    String text = "В русском тексте каждую букву заменить ее порядковым номером в \n" +
        "алфавите. При выводе в одной строке печатать текст с двумя\n" +
        "пробелами между буквами, в следующей строке внизу\n" +
        "под каждой буквой печатать ее номер.";
    System.out.println(text.replaceAll("pa", "po"));
}

```

```

В русском тексте каждую букву заменить ее порядковым номером в
алфавите. При выводе в одной строке печатать текст с двумя
пробелами между буквами, в следующей строке внизу
под каждой буквой печатать ее номер.

```

Рисунок 2 – Результат работы программы

### Задание:

2. Найти и напечатать, сколько раз повторяется в тексте каждое слово, которое встречается в нем.

### Ход работы:

#### Код программы:

```

public static void main(String[] args){
    String text = "В русском тексте каждую букву заменить ее порядковым номером в \n" +
        "алфавите. При выводе в одной строке печатать текст с двумя \n" +
        "пробелами между буквами, в следующей строке внизу \n" +
        "под каждой буквой печатать ее номер.";
    String [] words = text.toLowerCase().replaceAll("[.?!\\n]", "").split(" ");
    TreeMap<String, Integer> wordsTreeMap = new TreeMap<>();
    for (String word : words){
        Integer number = wordsTreeMap.get(word);
        if (number == null){
            number = 0;
        }
        number++;
        wordsTreeMap.put(word, number);
    }
    for (String word : wordsTreeMap.keySet()){
        System.out.println(word + ": " + wordsTreeMap.get(word));
    }
}

```

```

алфавите: 1
буквами: 1
буквой: 1
букву: 1
в: 4
внизу: 1
выводе: 1
двумя: 1
ее: 2
заменить: 1
каждой: 1
каждую: 1
между: 1
номер: 1
номерам: 1
одной: 1
печатать: 2
под: 1
порядковым: 1
пробелами: 1
при: 1
русском: 1
с: 1
следующей: 1
страке: 1
строке: 1
текст: 1
тексте: 1

```

Рисунок 3 – Результат работы программы

### Задание:

3. В тексте найти и напечатать  $n$  символов (и их количество), встречающихся наиболее часто.

### Ход работы:

#### Код программы:

```

public static void main(String[] args){
    String text = "В русском тексте каждую букву заменить ее порядковым номерам в \n" +
        "алфавите. При выводе в одной строке печатать текст с двумя \n" +
        "пробелами между буквами, в следующей строке внизу \n" +
        "под каждой буквой печатать ее номер.";
    int n;
    Scanner scanner = new Scanner(System.in);
    System.out.print("Введите количество символов: ");
    if (scanner.hasNextInt()) {
        n = scanner.nextInt();
    } else {
        System.out.println("error");
        n = -1;
    }
    char[] symbols = text.toLowerCase().replaceAll("[\n\\s]", "").toCharArray();
    HashMap<Character, Integer> symbolsHashMap = new HashMap<>();
    for (char sym : symbols){
        Integer number = symbolsHashMap.get(sym);
        if (number == null){
            number = 0;
        }
        number++;
        symbolsHashMap.put(sym, number);
    }
    symbolsHashMap.entrySet().stream()
        .sorted(Map.Entry.<Character, Integer>comparingByValue().reversed())
        .limit(n).forEach(System.out::println);
}

```

```

Введите количество символов: 5
e=20
a=14
в=13
т=12
о=12

```

Рисунок 4 – Результат работы программы

### Задание:

2. Найти наибольшее количество предложений текста, в которых есть одинаковые слова.

### Ход работы:

Код программы:

```

public static void main(String[] args){
    String text = ""
        В русском текст каждую букву заменить ее порядковым номером в\s
        алфавите. При выводе в одной строке печатать текст с двумя\s
        пробелами между буквами, в следующей строке внизу\s
        по каждой буквой печатать ее номер. Найти предложения\s
        с повторяющимися словами. текст"";
    String[] sentences = text.split("\\.");
    int max = 0, buf;
    String word_max = "", word_buf;
    for (String sentence : sentences){
        String[] words = sentence.toLowerCase(Locale.ROOT).split(" ");
        for (String word : words){
            buf = 0;
            word_buf = word;
            String[] sentences_eq = text.replace(sentence + "\\.", "").split("\\.");
            for (String sentence_eq : sentences_eq){
                String[] words_eq = sentence_eq.toLowerCase(Locale.ROOT).split(" ");
                for (String word_eq : words_eq){
                    if (word.equals(word_eq)){
                        buf++;
                        break;
                    }
                }
            }
            if (buf > max){
                max = buf;
                word_max = word_buf;
            }
        }
    }
    System.out.println(word_max + ": " + max);
}

```

```

текст: 3

```

Рисунок 5 – Результат работы программы

### Задание:

3. Найти такое слово в первом предложении, которого нет ни в одном из остальных предложений.

### Ход работы:

#### Код программы:

```
public static void main(String[] args){
    String text = ""
        В русском текст каждую букву заменить ее порядковым номером в\s
        алфавите. При выводе в русском одной строке печатать текст с двумя\s
        пробелами между буквами, в следующей строке внизу\s
        по каждой буквой печатать ее номер. Найти предложения\s
        с повторяющимися словами. "";
    String[] sentences = text.replaceAll(" ", "").split("\\.");
    String uniq_word = "Таких слов нет", buf;
    boolean flag = true;
    String[] words = sentences[0].toLowerCase(Locale.ROOT).split("\\s");
    sentences[0] = "";
    for (String word : words){
        buf = word;
        for (String sentence : sentences){
            if (sentence.contains(word)) {
                flag = false;
                break;
            }
        }
        if (flag){
            uniq_word = buf;
            break;
        }
        flag = true;
    }
    System.out.println(uniq_word);
}
```

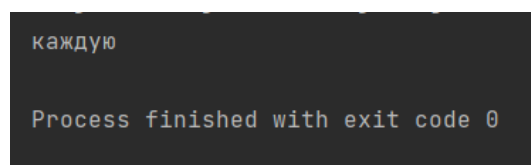


Рисунок 6 – Результат работы программы

### Задание:

2. Ввести текст и список слов. Для каждого слова из заданного списка найти, сколько раз оно встречается в тексте, и рассортировать слова по убыванию количества вхождений.

### Ход работы:

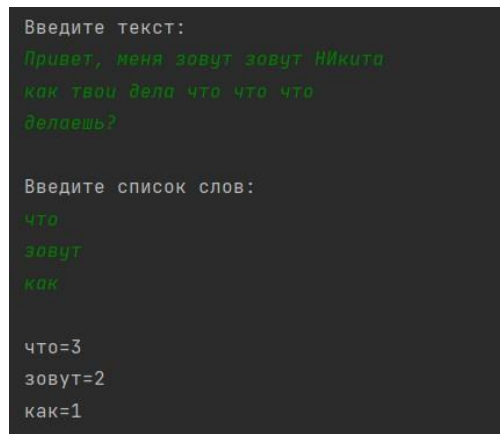
#### Код программы:

```
public static void main(String[] args){
    Scanner scanner = new Scanner(System.in);
```

```

System.out.println("Введите текст:");
String text = scanner.nextLine(), buf = " ";
while (!(buf.isEmpty())){
    buf = scanner.nextLine();
    text += "\n" + buf;
}
HashMap<String, Integer> wordsHashMap = new HashMap<>();
System.out.println("Введите список слов:");
buf = scanner.nextLine();
while (!(buf.isEmpty())){
    wordsHashMap.put(buf, 0);
    buf = scanner.nextLine();
}
String[] words = text.replaceAll(",", "").split("\\s");
for (String word : wordsHashMap.keySet()) {
    Integer number = wordsHashMap.get(word);
    for (String text_word : words) {
        if (Objects.equals(word, text_word)) {
            number++;
        }
    }
    wordsHashMap.put(word, number);
}
wordsHashMap.entrySet().stream()
    .sorted(Map.Entry.<String, Integer>comparingByValue().reversed())
    .forEach(System.out::println);
}

```



```

Введите текст:
Привет, меня зовут зовут Никита
как твои дела что что что
делаешь?

Введите список слов:
что
зовут
как

что=3
зовут=2
как=1

```

Рисунок 7 – Результат работы программы

### Задание:

3. Все слова текста рассортировать в порядке убывания их длин, при этом все слова одинаковой длины рассортировать в порядке возрастания в них количества гласных букв.

### Ход работы:

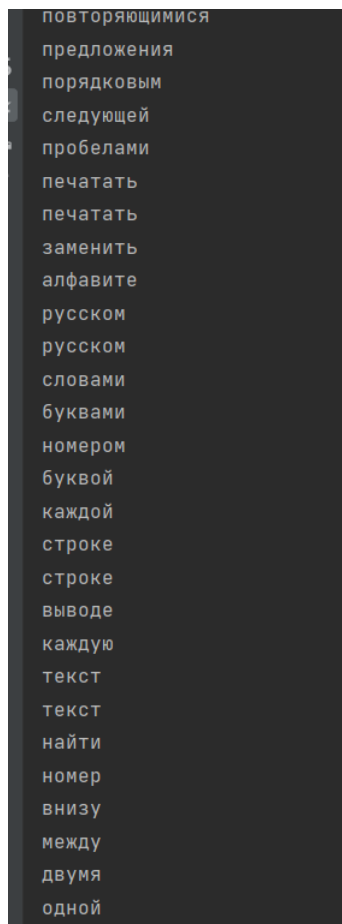
## Код программы:

```
public static void main(String[] args){
    String text = ""
        В русском текст каждую букву заменить ее порядковым номером в\s
        алфавите. При выводе в русском одной строке печатать текст с двумя\s
        пробелами между буквами, в следующей строке вниз\s
        по каждой буквой печатать ее номер. Найти предложения\s
        с повторяющимися словами. "";

    Comparator<String> comparator = new Comparator<String>() {
        @Override
        public int compare(String o1, String o2) {
            String alpha = "аеёиоуэюя";
            int buf_1, buf_2;
            if (o1.length() > o2.length()){
                return -1;
            } else if (o1.length() < o2.length()){
                return 1;
            } else{
                buf_1 = buf_2 = 0;
                for (char sym : o1.toCharArray()){
                    if (alpha.indexOf(sym) != -1){
                        buf_1++;
                    }
                }
                for (char sym : o2.toCharArray()){
                    if (alpha.indexOf(sym) != -1){
                        buf_2++;
                    }
                }
                if (buf_1 > buf_2){
                    return 1;
                } else return -1;
            }
        }
    };

    String[] words = text.toLowerCase(Locale.ROOT).replaceAll("[.\n]", "").split("\\s");
    List<String> words_list = new ArrayList<String>(Arrays.asList(words));
    words_list.sort(comparator);
    for (String word : words_list){
        System.out.println(word);
    }
}
```





повторяющимися  
предложения  
порядковым  
следующей  
пробелами  
печатать  
печатать  
заменить  
алфавите  
русском  
русском  
словами  
буквами  
номером  
буквой  
каждой  
строке  
строке  
выводе  
каждую  
текст  
текст  
найти  
номер  
внизу  
между  
двумя  
одной

Рисунок 8 – Результат работы программы

**Вывод:** лабораторная работа выполнена в соответствии с заданием и вариантом.