



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 09.04.01/12 Интеллектуальный анализ больших
данных в системах поддержки принятия решений.

О Т Ч Е Т

по лабораторной работе № 7

Вариант № 7

Название: строки, регулярные выражения

Дисциплина: языки программирования для работы с большими данными

Студент

ИУ6-23М

(Группа)

(Подпись, дата)

Ф.А. Лучкин

(И.О. Фамилия)

Преподаватель

(Подпись, дата)

П.В. Степанов

(И.О. Фамилия)

Москва, 2024

Цель: освоить принципы работы со строками и регулярными выражениями на языке программирования Java.

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.

Код класса Main:

```
package lab7_var1_7;

import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        int k = 3; // заданный индекс символа
        char symbol = 'a'; // символ для удаления или вставки

        System.out.println("Введите текст:");
        String text = scanner.nextLine();

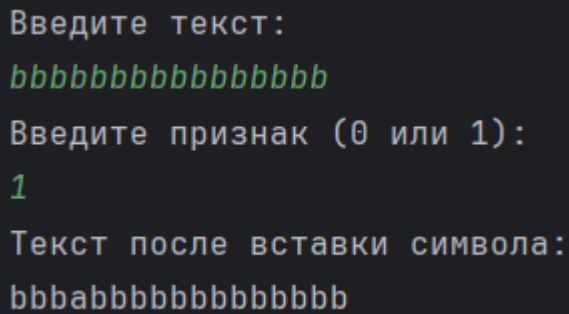
        System.out.println("Введите признак (0 или 1):");
        int flag = scanner.nextInt();

        if (flag == 0) {
            // Удалить символ везде, где он встречается
            String newText = text.replaceAll(String.valueOf(symbol), "");
            System.out.println("Текст после удаления символа:");
            System.out.println(newText);
        } else if (flag == 1) {
            // Вставить символ после k-го символа
            StringBuilder newText = new StringBuilder();
            for (int i = 0; i < text.length(); i++) {
                newText.append(text.charAt(i));
                if ((i + 1) == k) {
                    newText.append(symbol);
                }
            }
            System.out.println("Текст после вставки символа:");
            System.out.println(newText.toString());
        } else {
            System.out.println("Некорректный признак. Введите 0 или 1.");
        }
    }
}
```

Работа программы показана на рисунках 1-2.

[illegible]

Рисунок 1 – Работа программы



```
Введите текст:
bbbbbbbbbbbbbbbbbb
Введите признак (0 или 1):
1
Текст после вставки символа:
bbbabbbbbbbbbbbbbbb
```

Рисунок 2 – Работа программы

Задание 2: из небольшого текста удалить все символы, кроме пробелов, не являющиеся буквами. Между последовательностями подряд идущих букв оставить хотя бы один пробел.

Код класса Main:



```
package lab7_var1_8;

import java.util.Scanner;

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

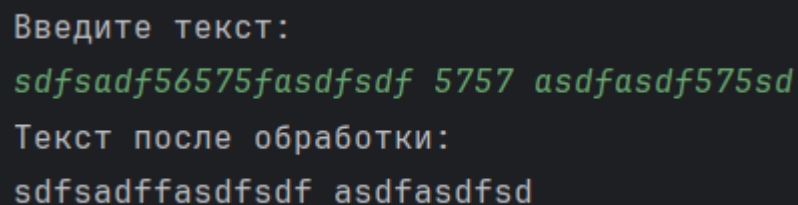
        System.out.println("Введите текст:");
        String text = scanner.nextLine();

        StringBuilder newText = new StringBuilder();
        boolean lastCharWasLetter = false;

        for (int i = 0; i < text.length(); i++) {
            char currentChar = text.charAt(i);
            if (Character.isLetter(currentChar)) {
                newText.append(currentChar);
                lastCharWasLetter = true;
            } else if (currentChar == ' ' && lastCharWasLetter) {
                newText.append(' ');
                lastCharWasLetter = false;
            }
        }

        System.out.println("Текст после обработки:");
        System.out.println(newText.toString());
    }
}
```

Работа программы показана на рисунке 3.



```
Введите текст:
sdfsadf56575fasdfsdf 5757 asdfasdf575sd
Текст после обработки:
sdfsadffasdfsdf asdfasdfsdf
```

Рисунок 3 – Работа программы

Задание 3: в тексте найти и напечатать все слова максимальной и все слова минимальной длины.

Код класса Main:

```
package lab7_var2_7;

import java.util.Scanner;

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

        System.out.println("Введите текст:");
        String text = scanner.nextLine();

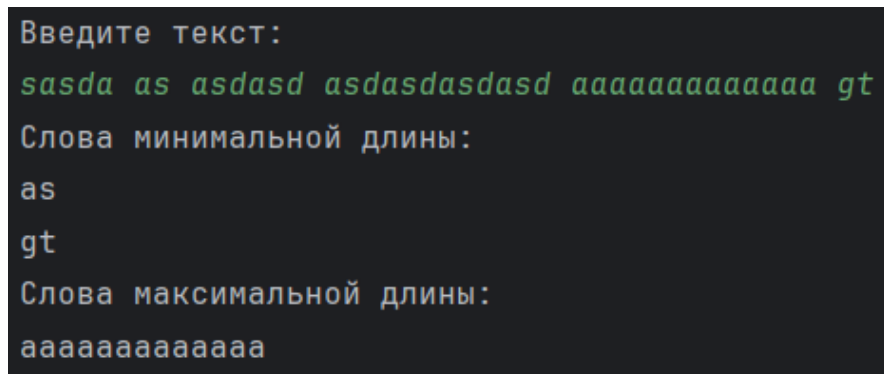
        String[] words = text.split("\\s+");
        int minLength = Integer.MAX_VALUE;
        int maxLength = 0;

        for (String word : words) {
            int length = word.length();
            if (length < minLength) {
                minLength = length;
            }
            if (length > maxLength) {
                maxLength = length;
            }
        }

        System.out.println("Слова минимальной длины:");
        for (String word : words) {
            if (word.length() == minLength) {
                System.out.println(word);
            }
        }

        System.out.println("Слова максимальной длины:");
        for (String word : words) {
            if (word.length() == maxLength) {
                System.out.println(word);
            }
        }
    }
}
```

Работа программы показана на рисунке 4.



```
Введите текст:
sasda as asdasd asdasdasdasd aaaaaaaaaaaaaa gt
Слова минимальной длины:
as
gt
Слова максимальной длины:
aaaaaaaaaaaaa
```

Рисунок 4 – Работа программы

Задание 4: напечатать квитанцию об оплате телеграммы, если стоимость одного слова задана.

Код модуля Main:

```
package lab7_var2_8;

import java.util.Scanner;

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

        System.out.println("Введите текст телеграммы:");
        String telegramText = scanner.nextLine();

        System.out.println("Введите стоимость одного слова:");
        double wordCost = scanner.nextDouble();

        String[] words = telegramText.split("\\s+");
        int wordCount = words.length;
        double totalCost = wordCount * wordCost;

        System.out.println("Квитанция об оплате телеграммы:");
        System.out.println("Текст телеграммы: " + telegramText);
        System.out.println("Количество слов: " + wordCount);
        System.out.println("Стоимость одного слова: " + wordCost);
        System.out.println("Общая стоимость: " + totalCost);
    }
}
```

Работа программы показана на рисунке 5.

```
Введите текст телеграммы:
lol kek cheburek lol aboba amogus
Введите стоимость одного слова:
6
Квитанция об оплате телеграммы:
Текст телеграммы: lol kek cheburek lol aboba amogus
Количество слов: 6
Стоимость одного слова: 6.0
Общая стоимость: 36.0
```

Рисунок 5 – Работа программы

Задание 5: 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.

Код модуля Main:

```
package lab7_var3_7;

import java.util.Scanner;

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

```

        System.out.println("Введите текст для шифрования:");
        String text = scanner.nextLine();

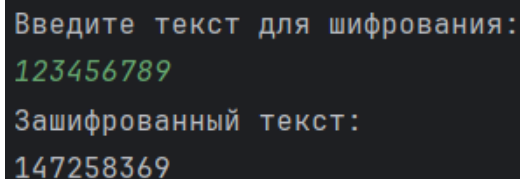
        StringBuilder encryptedText = new StringBuilder();

        for (int i = 0; i < text.length(); i += 3) {
            encryptedText.append(text.charAt(i));
        }
        for (int i = 1; i < text.length(); i += 3) {
            encryptedText.append(text.charAt(i));
        }
        for (int i = 2; i < text.length(); i += 3) {
            encryptedText.append(text.charAt(i));
        }

        System.out.println("Зашифрованный текст:");
        System.out.println(encryptedText.toString());
    }
}

```

Результат работы программы показан на рисунке 6.



```

Введите текст для шифрования:
123456789
Зашифрованный текст:
147258369

```

Рисунок 6 – Работа программы

Задание 6: на основании правила кодирования, описанного в предыдущем примере, расшифровать заданный набор символов.

Код модуля Main:

```

package lab7_var3_8;

import java.util.Scanner;

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

        System.out.println("Введите зашифрованный текст для расшифровки:");
        String encryptedText = scanner.nextLine();

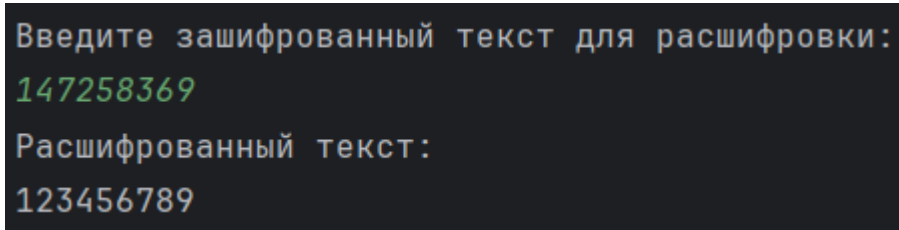
        int textLength = encryptedText.length();
        int groupSize = textLength / 3;
        StringBuilder decryptedText = new StringBuilder();

        for (int i = 0; i < groupSize; i++) {
            decryptedText.append(encryptedText.charAt(i));
            decryptedText.append(encryptedText.charAt(i + groupSize));
            decryptedText.append(encryptedText.charAt(i + 2 * groupSize));
        }

        System.out.println("Расшифрованный текст:");
        System.out.println(decryptedText.toString());
    }
}

```

Работа программы показана на рисунке 7.

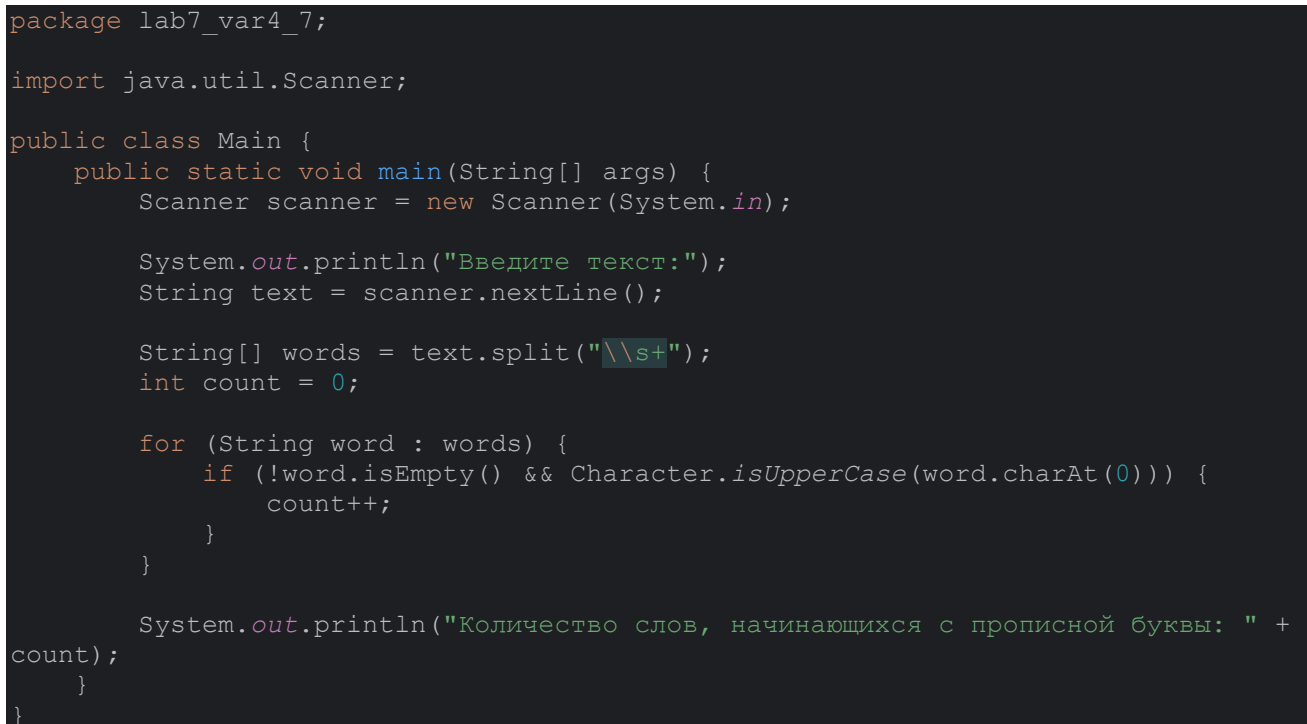


```
Введите зашифрованный текст для расшифровки:  
147258369  
Расшифрованный текст:  
123456789
```

Рисунок 7 – Работа программы

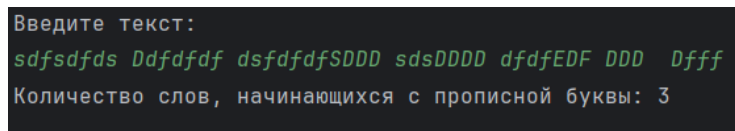
Задание 7: подсчитать, сколько слов в заданном тексте начинается с прописной буквы.

Код модуля Main:



```
package lab7_var4_7;  
  
import java.util.Scanner;  
  
public class Main {  
    public static void main(String[] args) {  
        Scanner scanner = new Scanner(System.in);  
  
        System.out.println("Введите текст:");  
        String text = scanner.nextLine();  
  
        String[] words = text.split("\\s+");  
        int count = 0;  
  
        for (String word : words) {  
            if (!word.isEmpty() && Character.isLowerCase(word.charAt(0))) {  
                count++;  
            }  
        }  
  
        System.out.println("Количество слов, начинающихся с прописной буквы: " +  
count);  
    }  
}
```

Результат работы программы показан на рисунке 8.

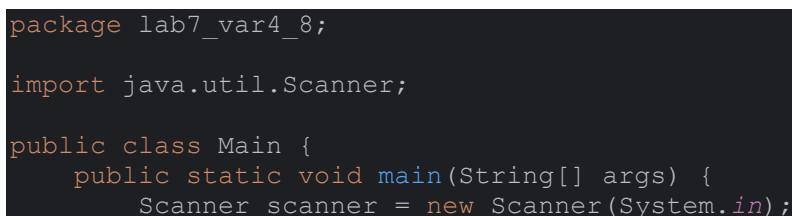


```
Введите текст:  
sdfsdfs Ddfdfdf dsfdfdfSDDD sdsDDDD dfdfEDF DDD Dfff  
Количество слов, начинающихся с прописной буквы: 3
```

Рисунок 8 – Работа программы

Задание 8: подсчитать, сколько раз заданное слово входит в текст.

Код модуля Main:



```
package lab7_var4_8;  
  
import java.util.Scanner;  
  
public class Main {  
    public static void main(String[] args) {  
        Scanner scanner = new Scanner(System.in);  
  
        String text = scanner.nextLine();  
        String word = scanner.nextLine();  
  
        int count = 0;  
        while (text.contains(word)) {  
            count++;  
            text = text.replace(word, "");  
        }  
  
        System.out.println("Количество вхождений слова: " + count);  
    }  
}
```

```

System.out.println("Введите текст:");
String text = scanner.nextLine();

System.out.println("Введите слово для подсчета:");
String wordToCount = scanner.next();

String[] words = text.split("\\s+");
int count = 0;

for (String word : words) {
    if (word.equalsIgnoreCase(wordToCount)) {
        count++;
    }
}

System.out.println("Слово \"" + wordToCount + "\" встречается " + count +
" раз(а) в тексте.");
}
}

```

Работа программы показана на рисунке 9.

```

Введите текст:
dfgsd lol sdfsdflol lol lol sdfstt lololol
Введите слово для подсчета:
lol
Слово "lol" встречается 3 раз(а) в тексте.

```

Рисунок 9 – Работа программы

Ссылка на git-репозиторий: https://github.com/FedorLuchkin/Java_bmstu

Вывод: были освоены принципы работы со строками и регулярными выражениями на языке программирования Java.