



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

Вариант 10

Дисциплина: Языки программирования для работы с большими данными

Преподаватель _____ П.В. Степанов
(Подпись, дата) (И.О. Фамилия)

Москва, 2022

Цель работы:

Получение навыков работы со строками и регулярными выражениями в Java.

Выполнение:

Задание 1:

1. В каждом слове текста k -ю букву заменить заданным символом. Если k больше длины слова, корректировку не выполнять.
2. Удалить из текста его часть, заключенную между двумя символами, которые вводятся (например, между скобками '(' и ')' или между звездочками '*' и т.п.).

Задание 2:

1. В тексте найти все пары слов, из которых одно является обращением другого.
2. В тексте найти первую подстроку максимальной длины, не содержащую букв.

Задание 3:

1. В тексте нет слов, начинающихся одинаковыми буквами. Напечатать слова текста в таком порядке, чтобы последняя буква каждого слова совпадала с первой буквой последующего слова. Если все слова нельзя напечатать в таком порядке, найти такую цепочку, состоящую из наибольшего количества слов.
2. Рассортировать слова русского текста по возрастанию доли гласных букв (отношение количества гласных к общему количеству букв в слове).

Задание 4:

1. Все слова английского текста рассортировать по возрастанию количества заданной буквы в слове. Слова с одинаковым количеством расположить в алфавитном порядке.
2. Исключить из текста подстроку максимальной длины, начинающуюся и заканчивающуюся одним и тем же символом.

Листинг файла WorkText.java

```
package com.company;

import java.util.ArrayList;
import java.util.Arrays;

public class WorkText {
    // encapsulation
    private static boolean isPolindromI(String a,String b)
    {
        // register is not important
        if(a.length()!=b.length())
            return false;
        int l=0;
        int r=b.length()-1;
        while(l<r)
        {
            if(Character.toLowerCase(a.charAt(l))!=Character.toLowerCase(b.charAt(r)))
                return false;
            l++;
            r--;
        }
        return true;
    }
    // Var1
    public static String changeKSymbolWord(String text,int k,char sym)
    {
        String []word=text.split("\\P{L}+");
        for(int i=0;i<word.length;i++)
        {
            if(word[i].length()>=k&&k>0)
            {
                word[i]=word[i].replace(word[i].charAt(k-1),sym);
            }
        }
        return String.join(" ",word);
    }
    // Var2
    public static void PairPolindrom(String text)
    {
        String []word=text.split("\\P{L}+");

        for(int i=0;i<word.length-1;i++)
        {
            // iterate over all pairs
            for(int j=i+1;j<word.length;j++)
            {
                if(isPolindromI(word[i],word[j]))
                    System.out.println(word[i]+"<->"+word[j]);
            }
        }
    }
    // Var3
    public static void CyclikWords(String text)
    {
        String []word=text.split("\\P{L}+");
        int mxCnt=0;
        int begPos=0; // word beginning
        int endPos=0; // word end
        for(int i=0;i<word.length-1;i++)
        {
            if(Character.toLowerCase(word[i].charAt(word[i].length()-1))==Character.toLowerCase(word[i+1].charAt(0)))
            {

```

```

        int Cnt=1;
        int bg=i;

while (i+1<word.length&&Character.toLowerCase(word[i].charAt(word[i].length()-
1))==Character.toLowerCase(word[i+1].charAt(0)))
    {
        Cnt++;
        i++;
    }
    if (Cnt>mxCnt)
    {
        // If the chain is max then save it
        mxCnt=Cnt;
        begPos=bg;
        endPos=i;
    }
    }

    for (int i=begPos; i<=endPos; i++)
    {
        System.out.print(word[i]+" ");
    }
}
// Var4
public static String SortWordBySym(String text,char sym)
{
    String []word=text.split("\\P{L}+");
    CompareString cm=new CompareString(sym); // ready comparator
    Arrays.sort(word,cm);
    return String.join(" ",word);
}
}

```

Листинг файла CompareString.java

```

package com.company;

import java.util.Comparator;

// Create comparator to compare via exact letter
public class CompareString implements Comparator<String> {
    private char ch;
    public CompareString(char ch)
    {
        this.ch=ch;
    }
    // Method which returns letter count in a specified string
    private static int getCntLetter(String s,char ch)
    {
        int ans=0;
        for (int i=0; i<s.length(); i++)
            if (s.charAt(i)==ch)
                ans++;
        return ans;
    }

    @Override
    public int compare(String o1, String o2) {
        return getCntLetter(o1,this.ch)-getCntLetter(o2,this.ch);
    }
}

```

Листинг файла Main.java

```

package com.company;

import java.util.Arrays;

```

```

import java.util.Collections;
import java.util.Scanner;

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

import java.util.Locale;
import java.util.logging.SocketHandler;

public class Main {
    public static void main(String[] args) {
        boolean quit=false;
        Scanner cin=new Scanner(System.in);
        while(!quit)
        {
            System.out.println("11 -Вариант 1, задание 1");
            System.out.println("110 -Вариант 1, задание 10");
            System.out.println("21 -Вариант 2, задание 1");
            System.out.println("210 -Вариант 2, задание 10");
            System.out.println("31 -Вариант 3, задание 1");
            System.out.println("310 -Вариант 3, задание 10");
            System.out.println("41 -Вариант 4, задание 1");
            System.out.println("410 -Вариант 4, задание 10");
            System.out.println("5 -Выход");
            int cmd=Integer.parseInt(cin.nextLine());
            switch (cmd)
            {
                case 11:
                {
                    System.out.println("Вариант 1 №1");
                    System.out.println("Введите текст: ");
                    String text= cin.nextLine(); // enter text
                    int k;
                    System.out.println("Введите k: ");
                    k = Integer.parseInt(cin.nextLine());
                    char ch;
                    System.out.println("Введите символ для замены: ");
                    ch=cin.nextLine().charAt(0);
                    System.out.println(WorkText.changeKSymbolWord(text,k,ch));
                    break;
                }
                case 110:
                {
                    System.out.println("Вариант 1 №10");
                    String text2 = "Hello, " +
                        "How (are you doing, " +
                        "I am fine, " +
                        ")I am done.";
                    StringBuilder str = new StringBuilder(text2);
                    System.out.print("Введите символы для удаления через пробел - ");
                    String[] in_word;
                    Scanner in2 = new Scanner(System.in);
                    in_word = in2.nextLine().split(" ");
                    System.out.println(Arrays.toString(in_word));
                    System.out.println(str);
                    int i1 = text2.indexOf(in_word[0]);
                    int i2 = text2.indexOf(in_word[1]);
                    System.out.println(text2.length());
                    System.out.println(i1+" "+i2);
                    for (int i =i1; i!=i2+1; i++){
                        // text2 = text2.(String.valueOf(text2.charAt(i)), "7"), ;
                        str.setCharAt(i, '7');
                    }
                    text2 = str.toString().replace("7", "");
                    System.out.println(text2);
                    break;
                }
            }
        }
    }
}

```

```

    }
    case 21:
    {
        System.out.println("Вариант 2 №1");
        System.out.println("Введите текст: ");
        String text= cin.nextLine();
        WorkText.PairPolindrom(text);
        break;
    }
    case 210:
    {
        System.out.println("\nВариант 2 №10");

        String[] alph = "a b c d e f g h i j k l m n o p q r s t u v w x y
z".split(" ");
        String text4 = "Hello, " +
            "How (are you doing, 2345 " +
            "I am 523 fine, " +
            ")I am done.";
        String[] words4 = text4.split(" ");
        String result = "";
        boolean str_res = false;
        for (int i=0; i!= words4.length; i++){
            for (int j=0; j!=words4[i].length(); j++){
                if
(!Arrays.asList(alph).contains(String.valueOf(words4[i].charAt(j)).toLowerCase(Locale.ROOT))) {
                    } else {
                        str_res = false;
                    }
                }
                if (str_res && words4[i].length() > result.length()){
                    result = words4[i];
                }
                str_res =true;
            }
            System.out.println(result);
            break;
        }
    case 31:
    {
        System.out.println("Вариант 3 №1");
        System.out.println("Введите текст: ");
        String text= cin.nextLine();
        WorkText.CyclikWords(text);
        System.out.println("\n");
        break;
    }
    case 310:
    {
        System.out.println("Вариант 3 №10");
        String[] alph6 = "a e i o u y".split(" ");
        String text6 = "Hello, " +
            "How (are you doing, 2345 " +
            "I am 523 fine, " +
            ")I am done.";
        String[] words6 = text6.split(" ");
        int[] words_mass = new int[words6.length];
        System.out.println(text6);
        for (int i = 0; i!=words6.length; i++){
            int gl_count=0;
            for (int j=0; j!=words6[i].length(); j++){
                if
(Arrays.asList(alph6).contains(String.valueOf(words6[i].charAt(j)).toLowerCase(Locale.ROOT))) {
                    gl_count+=1;
                }
            }
        }
    }
}

```

```

        }
        words_mass[i] = gl_count;
    }

    for (int i=0; i!=words_mass.length; i++){
        for (int j=0; j!=words_mass.length; j++){
            if (words_mass[i] < words_mass[j]){
                int cash = words_mass[i];
                words_mass[i] = words_mass[j];
                words_mass[j] = cash;

                String cashS = words6[i];
                words6[i] = words6[j];
                words6[j] = cashS;
            }
        }
    }
    System.out.println(Arrays.toString(words6));
    break;
}
case 41:
{
    System.out.println("Вариант 4 №1");
    System.out.println("Введите текст: ");
    String text= cin.nextLine();
    char sym;
    System.out.println("Введите букву: ");
    sym=cin.nextLine().charAt(0);
    System.out.println(WorkText.SortWordBySym(text, sym));
    break;
}
case 410:
{
    System.out.println("Вариант 4 №10");
    String text8 = "Hello,\n" +
        "How (are you doing, 2345 H\n" +
        "I am 523 fine, \n" +
        ")I am done.\n";
    String[] words8 = text8.split("\n");
    System.out.println(text8);
    System.out.println();
    int[] str_1 = new int[words8.length];
    String cash_str = "";
    for (int i =0 ; i!=words8.length; i++){
        if
(String.valueOf(words8[i].charAt(0)).toLowerCase(Locale.ROOT).equals(String.valueOf(wor
ds8[i].
        charAt(words8[i].length()-1)).toLowerCase(Locale.ROOT))){
            if (cash_str.length() < words8[i].length()) cash_str =
words8[i];
        }
    }
    System.out.println(text8.replace(cash_str, ""));
    break;
}
case 5:
{
    quit=true;
    break;
}
}
}
}
}
}

```

```

Вариант 1 №1
Введите текст:
jg4rg4g8 h94g gh 489 h4gh 984hg
Введите k:
3
Введите символ для замены:
q
jgqrg g h g gh h gh hg

```

Рисунок 1 - Результат выполнения кода решения задачи 1, подзадачи 1

```

Вариант 1 №10
Введите символы для удаления через пробел - ( )
[(, )]
Hello, How (are you doing, I am fine, )I am done.
49
11 38
Hello, How I am done.

```

Рисунок 2 - Результат выполнения кода решения задачи 1, подзадачи 2

```

Вариант 2 №1
Введите текст:
ты ыт так кат
ты<->ыт
так<->кат

```

Рисунок 3 - Результат выполнения кода решения задачи 2, подзадачи 1

```

Вариант 2 №10
2345

```

Рисунок 4 - Результат выполнения кода решения задачи 2, подзадачи 2

```

Вариант 3 №1
Введите текст:
how wolk ker ter okay
how wolk ker

```

Рисунок 5 - Результат выполнения кода решения задачи 3, подзадачи 1

```

Вариант 3 №10
Hello, How (are you doing, 2345 I am 523 fine, )I am done.
[2345, 523, I, am, How, )I, am, (are, fine,, Hello,, doing,, done., you]

```

Рисунок 6 - Результат выполнения кода решения задачи 3, подзадачи 2


```
Вариант 4 №1
Введите текст:
fu fukyf yuuf uykfkuf yfo o foufo
Введите букву:
u
yf f fu fukyf uf yuuf uykfkuf
```

Рисунок 7 - Результат выполнения кода решения задачи 4, подзадачи 1

```
Вариант 4 №10
Hello,
How (are you doing, 2345 H
I am 523 fine,
)I am done.

Hello,

I am 523 fine,
)I am done.
```

Рисунок 8 - Результат выполнения кода решения задачи 4, подзадачи 2

Ссылка на программное решение:

Программное решение представлено в репозитории распределённой системы управления версиями Git:

<https://github.com/Wingo11/BigDataLanguages/tree/Lab7/src>

Вывод:

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