

Министерство образования Республики Беларусь

Учреждение образования

«Брестский государственный технический университет»

Кафедра ИИТ

ОТЧЕТ

ЛАБОРАТОРНАЯ РАБОТА №2

Выполнил студент
группы АС-50:
Федорук Д. А.
Проверил:
Крощенко А. А.

Брест 2020

Задание 1

Напишите программу, выполняющую чтение текстовых данных из файла и их последующую обработку:

Вариант 9) Напишите программу, которая использует генерацию случайных чисел для создания предложений. Программа должна использовать 4 массива строк, называемые noun (существительные), adjective (прилагательные), verb (глаголы) и preposition (предлоги). Указанные массивы должны считываться из файла. Программа должна создавать предложение, случайно выбирая слова из каждого массива в следующем порядке: noun, verb, preposition, adjective, noun. Слова должны быть разделены пробелами. При выводе окончательного предложения, оно должно начинаться с заглавной буквы и заканчиваться точкой. Программа должна генерировать 20 таких предложений.

```
package com.company;

import java.io.BufferedReader;
import java.io.FileNotFoundException;
import java.io.FileReader;
import java.io.IOException;
import java.util.ArrayList;

public class Main {

    public static void main(String[] args) throws IOException {
        //чтение и запись в массив существительных
        BufferedReader readerNoun = new BufferedReader(new FileReader("E:\\noun.txt"));
        String str1;

        ArrayList<String> noun = new ArrayList<String>();
        while((str1 = readerNoun.readLine()) != null) {
            if (!str1.isEmpty()) {
                noun.add(str1);
            }
        }
        String[] nouns = noun.toArray(new String[0]);

        //чтение и запись в массив глаголов
        BufferedReader readerVerb = new BufferedReader(new FileReader("E:\\verb.txt"));
        String str2;

        ArrayList<String> verb = new ArrayList<String>();
        while((str2 = readerVerb.readLine()) != null) {
            if (!str2.isEmpty()) {
                verb.add(str2);
            }
        }
        String[] verbs = verb.toArray(new String[0]);

        //чтение и запись в массив предлогов
        BufferedReader readerPreposition = new BufferedReader(new FileReader("E:\\preposition.txt"));
        String str3;

        ArrayList<String> preposition = new ArrayList<String>();
        while((str3 = readerPreposition.readLine()) != null) {
            if (!str3.isEmpty()) {
                preposition.add(str3);
            }
        }
    }
}
```

```

String[] prepositions = preposition.toArray(new String[0]);

//чтение и запись в массив прилагательных

BufferedReader readerAdjective = new BufferedReader(new FileReader("E:\\adjective.txt"));
String str4;

ArrayList<String> adjective = new ArrayList<String>();
while((str4 = readerAdjective.readLine()) != null ) {
    if (!str4.isEmpty()) {
        adjective.add(str4);
    }
}
String[] adjectives = adjective.toArray(new String[0]);

//вывод предложений
for (int i=0;i<20;i++){
    int number1= (int) (Math.random()*10);
    int number2= (int) (Math.random()*10);
    int number3= (int) (Math.random()*10);
    int number4= (int) (Math.random()*10);
    int number5= (int) (10 + Math.random()*10);
    System.out.println(nouns[number1]+" "+verbs[number2]+" "+prepositions[number3]+"
"+adjectives[number4]+" "+nouns[number5]+".");
}
}
}

```

```

C:\Users\User\jdk\corretto-11.0.8\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2020.2.2\lib\idea_rt.jar=51666:C:\Program Files\JetBrains\IntelliJ IDEA 2020
Main
Main
Меч играть после осенний меч.
Ангел носить над покинутая победа.
Жизнь ломать по красивый ангел.
Меч ломать по одинокая ангел.
Ангел носить после одинокая человек.
Честь носить несмотря на одинокая свобода.
Свобода рубить к верный меч.
Чайник читать к яркий смерть.
Меч ходить к осенний человек.
Чайник ходить над разрисованная победа.
Победа играть в одинокая болото.
Честь рубить несмотря на яркий чайник.
Жизнь читать с красивый смерть.
Смерть менять в покинутая ангел.
Честь пилить над покинутая честь.
Честь ходить в красная чайник.
Жизнь рубить в разрисованная чайник.
Победа пилить по разрисованная человек.
Жизнь играть над покинутая свобода.
Чайник носить несмотря на чистый меч.

Process finished with exit code 0

```

Задание 2

Вариант 9) Утилита join объединяет строки двух упорядоченных текстовых файлов на основе наличия общего поля. По своему функционалу схоже с оператором JOIN, используемого в языке SQL для реляционных баз данных, но оперирует с текстовыми файлами. Команда join принимает на входе два текстовых файла и некоторое число аргументов. Если не передаются никакие аргументы командной строки, то данная команда ищет пары строк в двух файлах, обладающие совпадающим первым полем (последовательностью символов, отличных от пробела), и выводит строку, состоящую из первого поля и содержимого обоих строк. Ключами -1 или -2 задаются номера сравниваемых полей для первого и второго файла, соответственно. Если в качестве одного из файлов указано – (но не обоих сразу!), то в этом случае вместо файла считывается стандартный ввод. Формат использования: join [-1 номер_поля] [-2 номер_поля] файл1 файл2 [файл3] Параметры: • - 1 field_num Задаёт номер поля в строке для первого файла, по которому будет выполняться соединение. • - 2 field_num Задаёт номер поля в строке для второго файла, по которому будет выполняться соединение.

Аргументы: • файл1, файл2 – входные файлы • файл3 – выходной файл, куда записывается результат работы программы. Примеры использования: Пусть задан файл 1.txt со следующим содержимым:

1 abc

2 lmn

3 pqr

и файл 2.txt со следующим содержимым: 1 abc 3 lmn 9 orq

Тогда, выполнение команды join 1.txt 2.txt даст следующий результат: 1 abc abc 3 pqr lmn

Поскольку в обоих файлах есть строки, чьё первое поле совпадает (1, 3), выполнение команды join -1 2 -2 2 1.txt 2.txt даст результат abc 1 1 lmn 2 3 поскольку теперь сравнение выполняется по 2-му полю для первого и второго файла соответственно.

```
package com.company;

import java.io.FileWriter;
import java.io.IOException;
import java.io.PrintWriter;
import java.nio.charset.Charset;
import java.nio.file.Files;
import java.nio.file.Paths;
import java.util.Arrays;
import java.util.List;
import java.util.Scanner;

public class Main {

    public static void main(String[] args) throws IOException {
        List<String> firstFile = Files.readAllLines(Paths.get("File1.txt"), Charset.defaultCharset());
        List<String> secondFile = Files.readAllLines(Paths.get("File2.txt"), Charset.defaultCharset());

        FileWriter writer = new FileWriter("File3.txt");

        int first = Integer.parseInt(args[1]);
        int second = Integer.parseInt(args[2]);
```

```

int length = firstFile.size();
int length1 = 0;
int length2 = 0;

String str1;
String str2;

int index1;
int index2;

char[] st111 = new char[10];
char[] st121 = new char[10];
char[] st211 = new char[10];
char[] st221 = new char[10];

char[] st112 = new char[10];
char[] st122 = new char[10];
char[] st212 = new char[10];
char[] st222 = new char[10];

char[] st113 = new char[10];
char[] st123 = new char[10];
char[] st213 = new char[10];
char[] st223 = new char[10];

char[] st114 = new char[10];
char[] st124 = new char[10];
char[] st214 = new char[10];
char[] st224 = new char[10];

if (first == 1 && second == 1) {
    for (int i = 0; i < length; i++) {
        str1 = firstFile.get(i);
        str2 = secondFile.get(i);
        length1 = str1.length();
        length2 = str2.length();
        index1 = str1.indexOf(' ');
        index2 = str2.indexOf(' ');
        str1.getChars(0, index1, st111, 0);
        str2.getChars(0, index2, st211, 0);
        str1.getChars(index1, length1, st121, 0);
        str2.getChars(index2, length2, st221, 0);
        if (Arrays.equals(st111, st211)) {
            writer.write(st111);
            writer.write(st121);
            writer.write(st221);
            writer.write("\n");
        }
    }
}

if (first == 1 && second == 2) {
    for (int i = 0; i < length; i++) {
        str1 = firstFile.get(i);
        str2 = secondFile.get(i);
        length1 = str1.length();
        length2 = str2.length();
        index1 = str1.indexOf(' ');
        index2 = str2.indexOf(' ');
        str1.getChars(0, index1, st112, 0);
        str2.getChars(0, index2, st212, 0);
        str1.getChars(index1, length1, st122, 0);
    }
}

```

```

        str2.getChars(index2 + 1, length2, st222, 0);
        if (Arrays.equals(st112, st222)) {
            writer.write(st112);
            writer.write(st122);
            writer.write(st212);
            writer.write("\n");
        }
    }
}

if (first == 2 && second == 1) {
    for (int i = 0; i < length; i++) {
        str1 = firstFile.get(i);
        str2 = secondFile.get(i);
        length1 = str1.length();
        length2 = str2.length();
        index1 = str1.indexOf(' ');
        index2 = str2.indexOf(' ');
        str1.getChars(0, index1, st113, 0);
        str2.getChars(0, index2, st213, 0);
        str1.getChars(index1 + 1, length1, st123, 0);
        str2.getChars(index2, length2, st223, 0);
        System.out.println(st213);
        if (Arrays.equals(st123, st213)) {
            writer.write(st123);
            writer.write(st113);
            writer.write(st213);
            writer.write("\n");
        }
    }
}

if (first == 2 && second == 2) {
    for (int i = 0; i < length; i++) {
        str1 = firstFile.get(i);
        str2 = secondFile.get(i);
        //System.out.print(str1);
        length1 = str1.length();
        length2 = str2.length();
        index1 = str1.indexOf(' ');
        index2 = str2.indexOf(' ');
        str1.getChars(0, index1, st114, 0);
        str2.getChars(0, index2, st214, 0);
        str1.getChars(index1, length1, st124, 0);
        str2.getChars(index2, length2, st224, 0);
        //System.out.println(st224);
        if (Arrays.equals(st124, st224)) {
            System.out.print(st124);
            System.out.print(" ");
            System.out.print(st114);
            System.out.print(" ");
            System.out.print(st214);
            writer.write(st124);
            writer.write(st114);
            writer.write(st214);
            writer.write("\n");
        }
    }
}

writer.close();

Scanner scanner = new Scanner(System.in);
String s = scanner.nextLine();

```

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

```
C:\Users\User\.jdk\openjdk-15.0.1\bin\java.exe "-javaagent:D:\IntelliJ IDEA Community Edition 2020.1.1\lib\idea_rt.jar=58728:D:\IntelliJ IDEA Community Edition 2020.1.1\bin" -Dfile.encoding=UTF-8
dsfg 3 3
```