

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
“БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ”
Кафедра ИИТ

ОТЧЁТ
по лабораторной работе №2
за 1 семестр 3 курса

Выполнил:
студент группы ПО-9(1)
3 курса
Зейденс Никита
Вячеславович

Проверил:
Крощенко
А. А.

Брест, 2024

Цель: Приобрести базовые навыки работы с файловой системой в Java.

Вариант 5

Задание 1: Напишите программу, выполняющую чтение текстовых данных из файла и печатает таблицу, показывающую, сколько раз в этом тексте встречаются однобуквенные слова, двухбуквенные слова, трёхбуквенные слова и т.д.

Код программы:

```
import java.io.*;
import java.util.HashMap;
import java.util.Scanner;
import java.util.Map;

public class Main {
    public static String readFile(String filename) {
        try(FileReader reader = new FileReader(filename))
        {
            int c;
            String str = "";
            while((c=reader.read())!=-1) {
                str += (char) c;
            }
            System.out.print(str + "\n");
            return str;
        }
        catch(IOException ex){
            System.out.println(ex.getMessage());
        }
        return "";
    }

    public static Map<Integer, Integer> tableOfWords(String str) {
        Map<Integer, Integer> table = new HashMap<>();
        String cleanedInput = str.replaceAll("[^a-zA-Z]+", " "); //Очистка строки
        String[] substrings = cleanedInput.split(" ");
        for(int i = 0; i < substrings.length; ++i) {
            int numberOfLetters = substrings[i].length();
            if(table.containsKey(numberOfLetters)) {
                int value = table.get(substrings[i].length());
                ++value;
                table.put(numberOfLetters, value);
            }
            else {
                table.put(numberOfLetters, 1);
            }
        }
        return table;
    }

    public static void main(String[] args) {
        Scanner in = new Scanner(System.in);
        System.out.print("Enter filename without extension: ");
        String fileName = in.next();
        Map<Integer, Integer> table = tableOfWords(readFile(fileName));
        for(Map.Entry<Integer, Integer> entry : table.entrySet()) {
            System.out.print("Number of letters: " + entry.getKey() + " Number of
words with this number of letters: " + entry.getValue() + "\n");
        }
    }
}
```

Результат работы:

```
Enter filename without extension: 10_w.txt
semper viverra nam libero justo laoreet sit amet cursus sit
Number of letters: 3 Number of words with this number of letters: 3
Number of letters: 4 Number of words with this number of letters: 1
Number of letters: 5 Number of words with this number of letters: 1
Number of letters: 6 Number of words with this number of letters: 3
Number of letters: 7 Number of words with this number of letters: 2

Enter filename without extension: 10_w.txt
semper viverra nam libero justo laoreet sit amet cursus sit semper viverra nam libero justo laoreet sit amet cursus sit
Number of letters: 3 Number of words with this number of letters: 6
Number of letters: 4 Number of words with this number of letters: 2
Number of letters: 5 Number of words with this number of letters: 2
Number of letters: 6 Number of words with this number of letters: 6
Number of letters: 7 Number of words with this number of letters: 4
```

Задание 2: Написать консольную утилиту, обрабатывающую ввод пользователя и дополнительные ключи. Проект упаковать в jar-файл, написать bat-файл для запуска.

Утилита `sr` осуществляет копирование файла из одного каталога в другой. Исходный файл остаётся неизменным, имя созданного файла может быть таким же, как у исходного, или измениться.

Формат использования: `sr [-f][-i][-n] исходный_файл целевой_файл`

- `-f` Разрешает удаление целевого файла, в который производится копирование, если он не может быть открыт для записи.
- `-i` Утилита будет запрашивать, следует ли перезаписывать конечный файл, имя которого совпадает с именем исходного. Для того, чтобы перезаписать файл, следует ввести `y` или его эквивалент. Ввод любого другого символа приведёт к отмене перезаписи данного файла
- `-n` Не перезаписывать существующий файл (отменяет предыдущий параметр `-i`)

Пример использования: `sr -fn src.txt dest.txt`

Копирует содержимое из `src.txt` в `dest.txt` с ключами `-f` и `-n`

Код программы:

```
import java.io.*;
import java.nio.file.Files;
import java.util.Scanner;

import static java.nio.file.StandardCopyOption.REPLACE_EXISTING;

public class Main {
    public static void copyWithParams(String[] args) {
        if(args[1].length() < 2 || args[1].length() > 4) {
            System.out.print("Wrong number of parameters, expected: 1-3
parameters.\n");
            return;
        }
        if(args[1].charAt(0) != '-') {
            System.out.print("Wrong parameters input form, expected: -
parameters\n");
            return;
        }
        boolean forceRewrite = false, askRewrite = false, noRewrite = false;
```

```

for(int i = 1; i < args[1].length(); ++i) {
    char symbol = args[1].charAt(i);
    if(symbol == 'f' && !forceRewrite)
    {
        forceRewrite = true;
        continue;
    }
    if(symbol == 'i' && !askRewrite)
    {
        askRewrite = true;
        continue;
    }
    if(symbol == 'n' && !noRewrite)
    {
        noRewrite = true;
        continue;
    }
    System.out.print("Unknown parameter, expected: 'f', 'i', 'n' as
parameters.\n");
    return;
}
String sourceFilePath = args[args.length - 2];
File sourceFile = new File(sourceFilePath);
if(!sourceFile.exists())
{
    System.out.print("File " + args[args.length - 2] + " doesn't exist");
    return;
}
String destinationFilePath = args[args.length - 1];
File destinationFile = new File(destinationFilePath);
if(destinationFile.exists() && forceRewrite)
{
    destinationFile.delete();
    System.out.print("File " + destinationFilePath + " deleted
successfully.\n");
    copy(args, false);
    return;
}
if(destinationFile.exists() && askRewrite && !noRewrite)
{
    System.out.print("Do you want to rewrite file? (y/n) ");
    Scanner in = new Scanner(System.in);
    String fileName = in.next();
    copy(args, fileName.equalsIgnoreCase("y"));
    return;
}
copy(args, false);
}
public static void copy(String[] args, boolean isRewritingNeeded) {
    File sourceFile = new File(args[args.length - 2]);
    File destinationFile = new File(args[args.length - 1]);
    try {
        if(isRewritingNeeded)
        {
            System.out.print("Copying with rewrite.\n");
            Files.copy(sourceFile.toPath(), destinationFile.toPath(),
REPLACE_EXISTING);
            System.out.println("File rewritten successfully.");
        }
        else
        {
            System.out.print("Copying without rewrite.\n");
            Files.copy(sourceFile.toPath(), destinationFile.toPath());
            System.out.println("File copied successfully.");
        }
    }
}

```

```

        } catch (IOException e) {
            System.out.println("Error while copying file: " + e.getMessage());
        }
    }

    public static void start(String[] args) {
        if(args.length < 3 || args.length > 4)
        {
            System.out.print("Wrong number of arguments, expected: 3-4
arguments.\n");
            return;
        }
        if(!args[0].equals("cp")) {
            System.out.print("Unknown command, expected: 'cp'.\n");
            return;
        }
        if(args.length == 3)
        {
            copy(args, false);
        }
        else
        {
            copyWithParams(args);
        }
    }

    public static void main(String[] args) {
        start(args);
        System.out.println("Press any key to continue...");
        Scanner scanner = new Scanner(System.in);
        scanner.nextLine();
        scanner.close();
    }
}

```

Результат работы:

Аргументы: cp -in 1.txt 2.txt

```

Copying without rewrite.
Error while copying file: 2.txt
Press any key to continue...

```

Аргументы: cp -i 1.txt 2.txt

```

Do you want to rewrite file? (y/n) y
Copying with rewrite.
File rewritten successfully.
Press any key to continue...

```

Аргументы: cp -i 1.txt 2.txt

```

File 2.txt deleted successfully.
Copying without rewrite.
File copied successfully.
Press any key to continue...

```

Вывод: Базовые навыки работы с файловой системой в Java были приобретены.