

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
“БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ”
КАФЕДРА ИНТЕЛЛЕКТУАЛЬНЫХ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

Отчёт
по лабораторной работе №2

Выполнил:
студент группы ПО-9
Ступак Д.Р

Проверил:
Крощенко А. А.

Брест 2024

Вариант 7

Цель работы: приобрести практические навыки обработки параметров командной строки, закрепить базовые знания языка программирования Java при решении практических задач

Задание 1

Необходимо подсчитать число цифр в текстовом файле. Локализовать и вывести на экран строку, содержащую цифру с порядковым номером $n/2$, где n – общее количество подсчитанных цифр.

Код программы:

```
import java.io.File;
import java.io.FileNotFoundException;
import java.util.*;

public class Task1 {
    public static void main(String[] args) throws FileNotFoundException {
        File file = new File("src/a.txt");
        Scanner scanner = new Scanner(file);
        ArrayList<String> lines = new ArrayList<>();
        while (scanner.hasNextLine()) {
            String line = scanner.nextLine();
            lines.add(line);
        }
        ArrayList<Integer> indexes = new ArrayList<>();
        for (int i = 0; i < lines.size(); i++){
            for (char c : lines.get(i).toCharArray()){
                if (Character.isDigit(c)){
                    indexes.add(i);
                }
            }
        }
        System.out.println(lines.get(indexes.get(indexes.size()/2)));
    }
}
```

Текстовый файл

```
20 (двадцать) – это натуральное число.
20 – это элемент множества N.
Это выражение можно записать так: 20N, а прочитать так: «двадцать принадлежит эн».
```

Результат работы:

```
20 – это элемент множества N.

Process finished with exit code 0
```

Задание 2

Утилита `uniq` отфильтровывает повторяющиеся строки во входном файле. Если входной файл

задан как `-` или не задан вовсе, то чтение производится из стандартного ввода. Если выходной

файл не задан, запись производится в стандартный вывод. Если одна и та же строка встречается

второй и более разы, то она не записывается в вывод программы.

Формат использования: `uniq [-c | -d | -u] [-i] [входной_файл [выходной_файл]]`, где

ключи имеют следующее значение:

- `-u` Выводить только те строки, которые не повторяются на входе.
- `-d` Выводить только те строки, которые повторяются на входе.
- `-c` Перед каждой строкой выводить число повторений этой строки на входе и один пробел.
- `-i` Сравнивать строки без учёта регистра.

Код программы

```
import java.io.*;
```

```
import java.util.*;
```

```
public class Task2 {  
    public static void main(String[] args) throws Exception {
```

```
        boolean count = false;
```

```
        boolean duplicates = false;
```

```
        boolean unique = false;
```

```
        boolean ignoreCase = false;
```

```
        String inputFile = null;
```

```
        String outputFile = null;
```

```
        for (int i = 1; i < args.length; i++) {
```

```
            switch (args[i]) {
```

```
                case "-c":
```

```
                    count = true;
```

```
                    break;
```

```
                case "-d":
```

```
                    duplicates = true;
```

```
                    break;
```

```
                case "-u":
```

```
                    unique = true;
```

```
                    break;
```

```
                case "-i":
```

```
                    ignoreCase = true;
```

```
                    break;
```

```
            default:
```

```
                if (inputFile == null) {
```

```
                    inputFile = args[i];
```

```
                } else if (outputFile == null) {
```

```
                    outputFile = args[i];
```

```
                }
```

```
                break;
```

```
            }
```

```
        }
```

```

try {
    BufferedReader reader;
    PrintWriter writer;

    if (inputFile != null) {
        reader = new BufferedReader(new FileReader(inputFile));
    } else {
        reader = new BufferedReader(new InputStreamReader(System.in));
    }

    if (outputFile != null) {
        writer = new PrintWriter(outputFile);
    } else {
        writer = new PrintWriter(System.out);
    }

    Map<String, Integer> countMap = new HashMap<>();

    String line;
    while ((line = reader.readLine()) != null) {
        if (ignoreCase) {
            line = line.toLowerCase();
        }

        if (count || duplicates || unique) {
            countMap.put(line, countMap.getOrDefault(line, 0) + 1);
        } else {
            writer.println(line);
        }
    }

    if (count || duplicates || unique) {
        for (Map.Entry<String, Integer> entry : countMap.entrySet()) {
            if (count){
                writer.println(entry.getValue() + " " + entry.getKey());
            } else if (duplicates && entry.getValue() > 1){
                writer.println(entry.getKey());
            } else if (unique && entry.getValue() == 1){
                writer.println(entry.getKey());
            }
        }
    }

    reader.close();
    writer.close();
} catch (IOException e) {
    System.err.println("Error: " + e.getMessage());
    System.exit(1);
}
}

```

Результат работы программы:

Вывод только уникальных строк

AA

BB

CC

CC

AA

BB