

Міністерство освіти і науки України

Національний університет «Львівська політехніка»

**Кафедра ЕОМ**



**Звіт**

до лабораторної роботи №7

з дисципліни: «Кросплатформенні засоби програмування»

На тему: «ПАРАМЕТРИЗОВАНЕ ПРОГРАМУВАННЯ »

Виконав: ст. гр. КІ-36

Лабенський В.В.

Прийняв:

Іванов Ю.С.

**Львів 2022**

## Лабораторна робота №7

**Мета:** оволодіти навиками параметризованого програмування мовою Java.

### Варіант №13

13.Словник

**Код програми:**

**Файл Vocabulary.java**

```
import java.util.*;

public class Vocabulary<T extends Data> {

    private final List<T> words;
    private final List<T> interpretation;

    public Vocabulary(){

        words = new ArrayList<T>();
        interpretation = new ArrayList<T>();
    }

    public void addLine(T a, T b)
    {

        words.add(a);
        interpretation.add(b);
    }

    public void deleteLine(int index){
        words.remove(index-1);
        interpretation.remove(index-1);
    }

    public void showVocabulary(){
        System.out.println("Vocabulary: ");

        for(int i = 0;i < words.size();i++){
            System.out.println((i+1)+". "+"Word: " +
words.get(i).getWord()+"\t-\t"+ interpretation.get(i).getWord());
        }
        System.out.println("\n");
    }
}
```

```

    }

    public void findMax(){
        int maxIndex1 = 0;
        int maxIndex2 = 0;
        int maxSize1=0;
        int maxSize2=0;

        for(int i = 0; i < words.size();i++){
            if(words.get(i).getSize() > maxSize1){
                maxIndex1 = i+1;
                maxSize1 = words.get(i).getSize();
            }
            if(interpretation.get(i).getSize() > maxSize2){
                maxIndex2 = i+1;
                maxSize2 = interpretation.get(i).getSize();
            }
        }
        System.out.println("Max Length of vocabulary words in line:
"+maxIndex1);
        System.out.println("Max Length of vocabulary interpretation in
line: "+maxIndex2+"\n");

    }
    public void a(){
        System.out.println(words.size());
    }
}

```

### Файл EnglishWord.java

```

public class WordEnglish implements Data{

    private String word;
    private int Length;

    public WordEnglish(String word) {
        this.word = word;
        Length = word.length();
    }
    @Override
    public int getSize() {

```

```

        return Length;
    }

    @Override
    public void print() {
        System.out.println(word);
    }

    @Override
    public int compareTo(Data o) {
        Integer s= Length;
        return s.compareTo(o.getSize());
    }

    @Override
    public String getWord() {
        return word;
    }
}

```

### Файл GermanWord.java

```

public class WordGerman implements Data {
    private String word;
    private int Length;

    public WordGerman(String word) {
        this.word = word;
        Length = word.length();
    }

    @Override
    public int getSize() {
        return Length;
    }

    @Override
    public void print() {
        System.out.println(word);
    }

    @Override
    public int compareTo(Data o) {
        Integer s= Length;

```

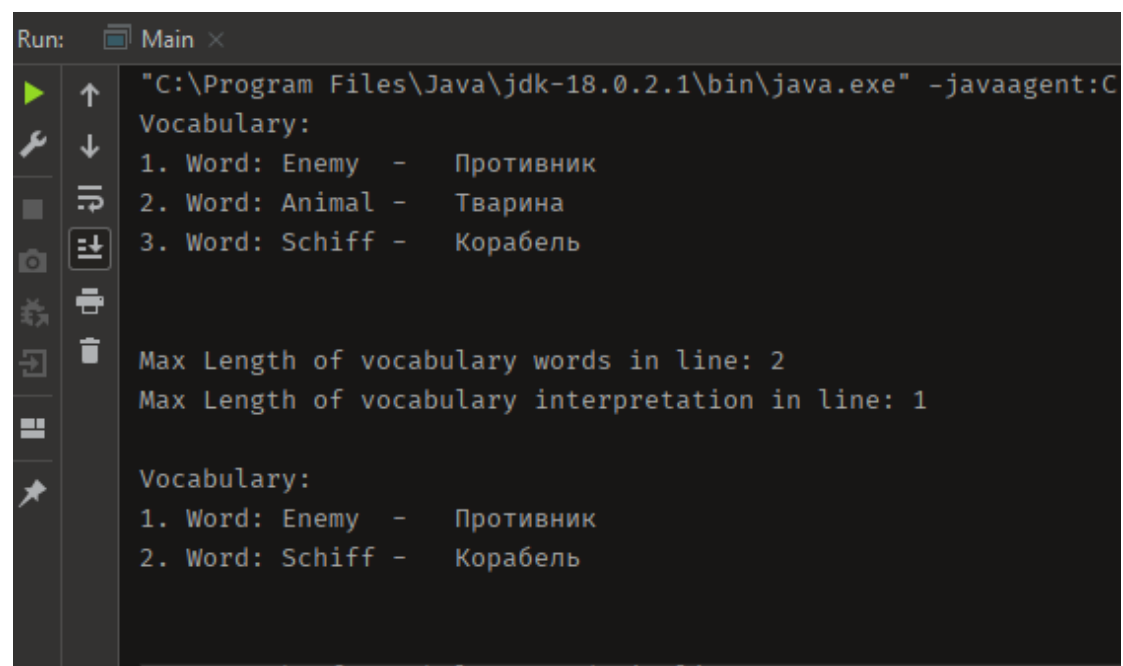
```

        return s.compareTo(o.getSize());
    }

    @Override
    public String getWord() {
        return word;
    }
}

```

**Приклад виконання програми:**



The screenshot shows the 'Run' console of an IDE. The command executed is: `"C:\Program Files\Java\jdk-18.0.2.1\bin\java.exe" -javaagent:C`. The output of the program is as follows:

```

Vocabulary:
1. Word: Enemy - Противник
2. Word: Animal - Тварина
3. Word: Schiff - Корабель

Max Length of vocabulary words in line: 2
Max Length of vocabulary interpretation in line: 1

Vocabulary:
1. Word: Enemy - Противник
2. Word: Schiff - Корабель

```

**Висновок:** На даній лаборатоній роботі я оволодів навиками параметризованого програмування мовою Java.