Міністерство освіти і науки України

Національний університет “Львівська політехніка”

Кафедра ЕОМ



**Звіт**

З лабораторної роботи №6

Варіант – 24

З дисципліни: «Кросплатформні засоби програмування»

На тему: «Файли у Java»

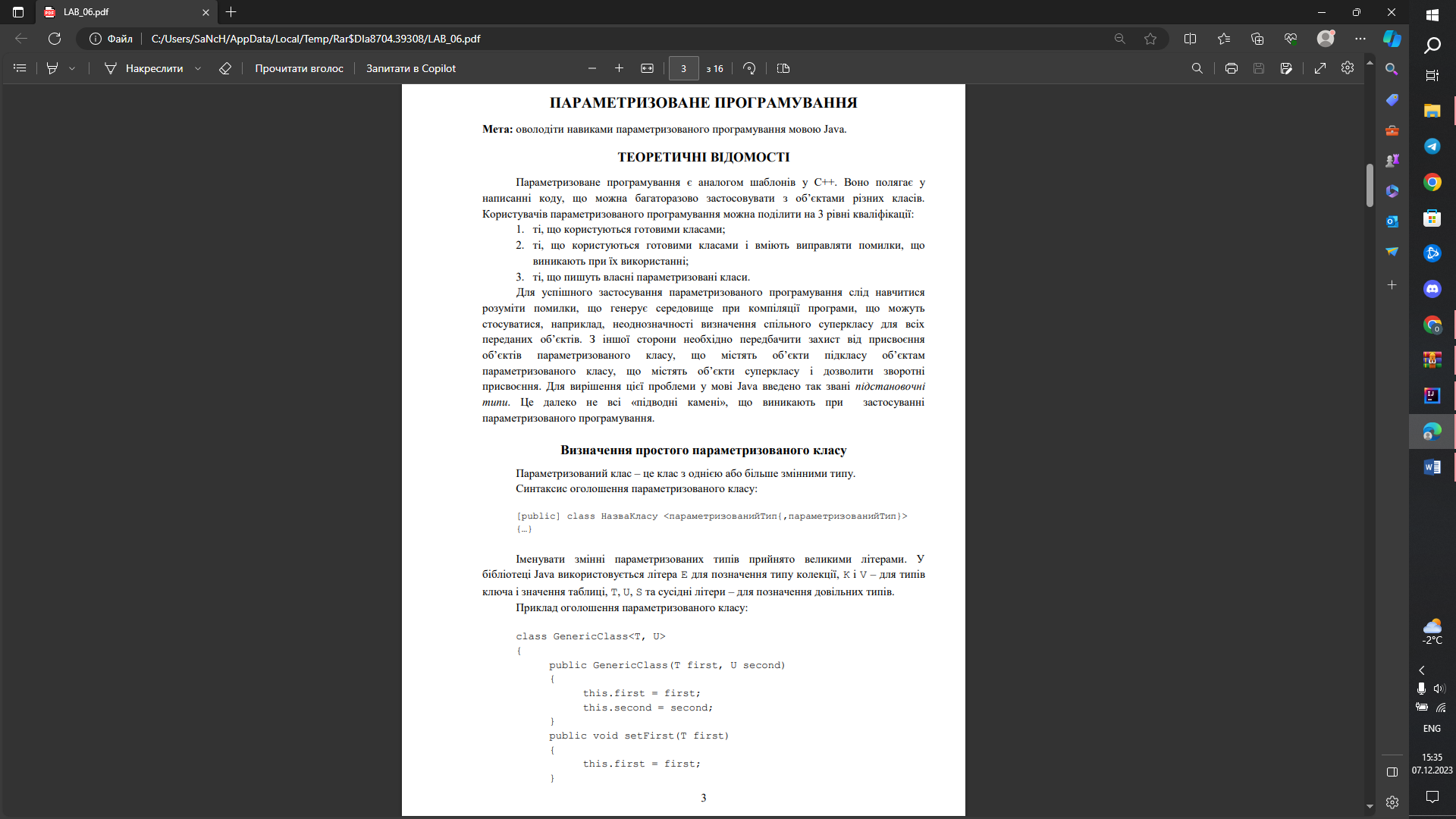
Виконав: ст. гр. КІ-301

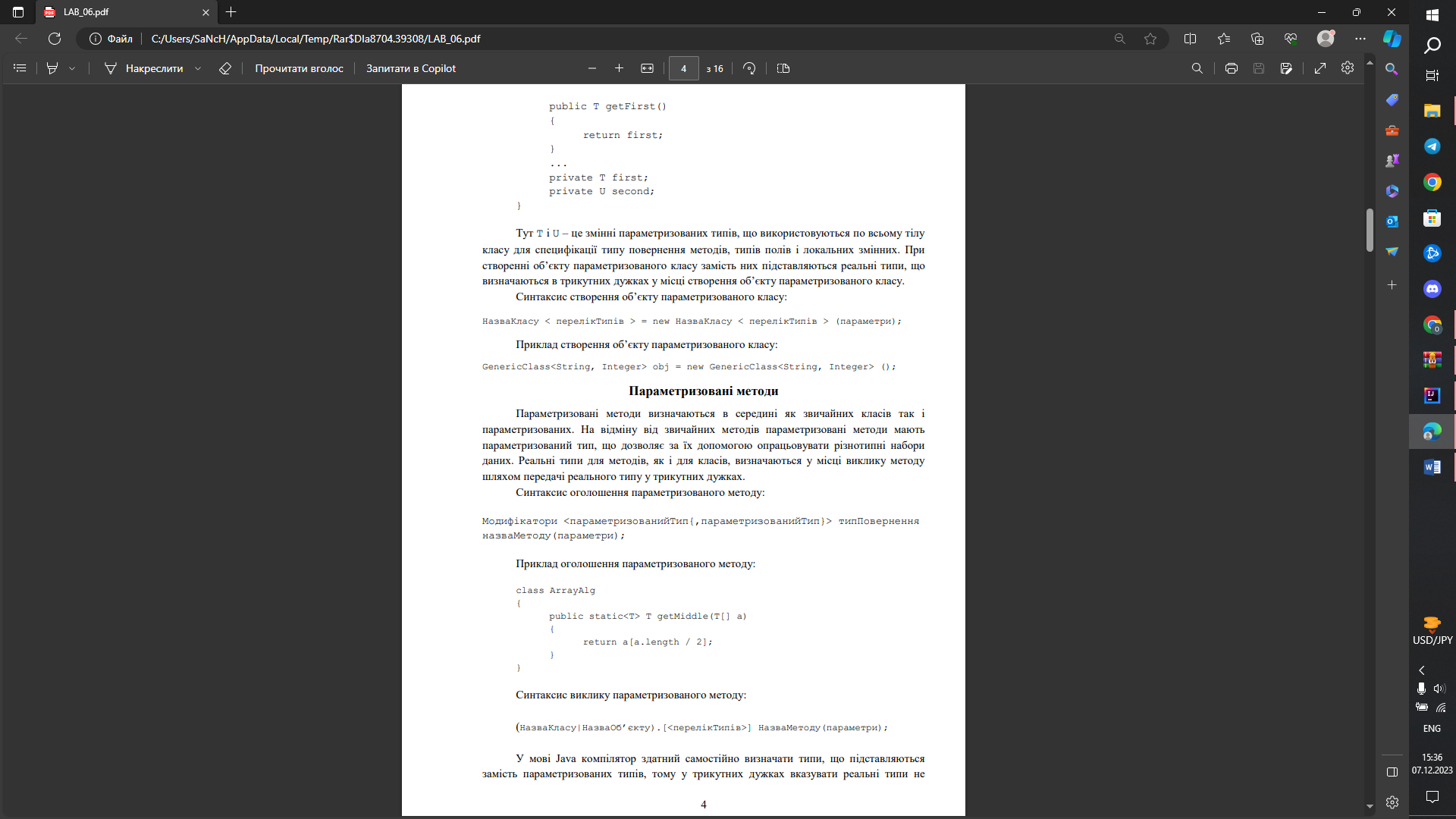
Чоланюк О.Ю.

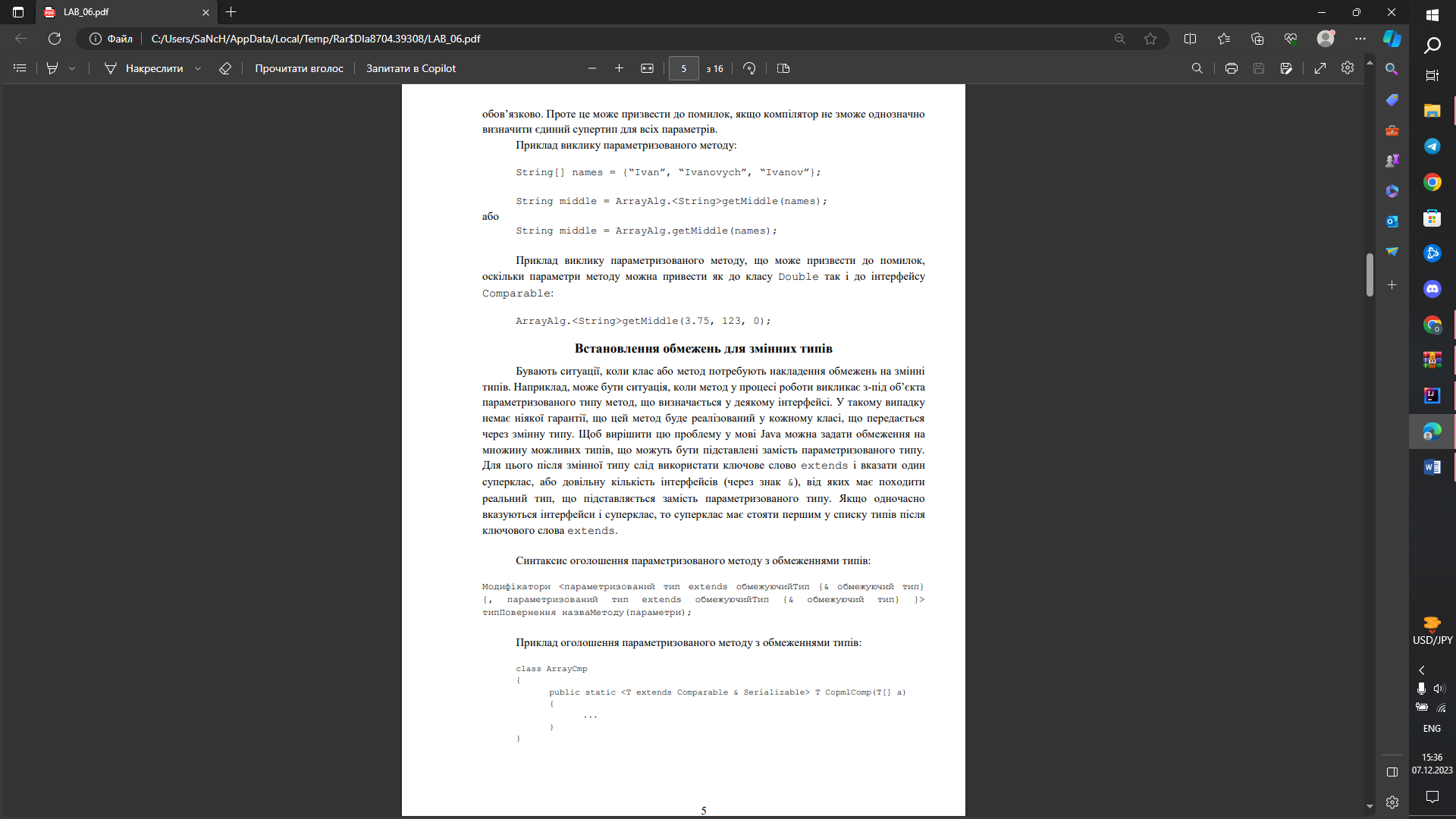
Прийняв:

Майдан М.В.

Львів 2023







ЗАВДАННЯ

1. Створити параметризований клас, що реалізує предметну область задану варіантом. Клас має містити мінімум 4 методи опрацювання даних включаючи розміщення та виймання елементів. Парні варіанти реалізують пошук мінімального елементу, непарні – максимального. Написати на мові Java та налагодити програму-драйвер для розробленого класу, яка мстить мінімум 2 різні класи екземпляри яких розмішуються у 9 екземплярі розробленого класу-контейнеру. Програма має розміщуватися в пакеті Група.Прізвище.Lab6 та володіти коментарями, які дозволять автоматично згенерувати документацію до розробленого пакету.

2. Автоматично згенерувати документацію до розробленого пакету.

3. Завантажити код на GitHub згідно методичних вказівок по роботі з GitHub.

4. Скласти звіт про виконану роботу з приведенням тексту програми, результату її виконання та фрагменту згенерованої документації та завантажити його у ВНС. 5. Дати відповідь на контрольні запитання.

Варіант 24: коробка з інструментами

package KZP\_Lab6\_Cholaniuk;  
  
public class Main {  
 public static void main(String[] args) {  
 ToolBox<? super Tool> toolBox = new ToolBox<Tool>();  
 toolBox.addTool(new Hammer("Молоток", "Метал", 500));  
 toolBox.addTool(new Screwdriver("Викрутка", "Пластик", 150));  
 toolBox.addTool(new Wrench("Розвідний ключ", "Сталь", 300));  
 Tool maxTool = toolBox.findMax();  
 System.*out*.print("Найкращий інструмент у наборі інструментів: \n");  
 maxTool.print();  
 }  
}

package KZP\_Lab6\_Cholaniuk;  
  
import java.util.ArrayList;  
  
public class ToolBox<T extends Tool> {  
 private ArrayList<T> tools;  
  
 public ToolBox() {  
 tools = new ArrayList<T>();  
 }  
 public T findMax() {  
 if (!tools.isEmpty()) {  
 T max = tools.get(0);  
 for (int i = 1; i < tools.size(); i++) {  
 if (tools.get(i).compareTo(max) > 0)  
 max = tools.get(i);  
 }  
 return max;  
 }  
 return null;  
 }  
 public void addTool(T tool) {  
 tools.add(tool);  
 System.*out*.print("Інструмент додано: ");  
 tool.print();  
 }  
 public void deleteTool(int i) {  
 tools.remove(i);  
 }  
}

package KZP\_Lab6\_Cholaniuk;  
public abstract class Tool implements Comparable<Tool> {  
 private String name;  
 private String material;  
 private int weight;  
 public Tool(String name, String material, int weight) {  
 this.name = name;  
 this.material = material;  
 this.weight = weight;  
 }  
 public String getName() {  
 return name;  
 }  
 public String getMaterial() {  
 return material;  
 }  
 public int getWeight() {  
 return weight;  
 }  
 public abstract void print();  
  
 public int compareTo(Tool tool) {  
 Integer w = weight;  
 return w.compareTo(tool.getWeight());  
 }  
}

package KZP\_Lab6\_Cholaniuk;  
public class Hammer extends Tool {  
 public Hammer(String name, String material, int weight) {  
 super(name, material, weight);  
 }  
 @Override  
 public void print() {  
 System.*out*.println("Молоток: " + getName() + ", Матеріал: " + getMaterial() + ", Вага: " + getWeight() + " г");  
 }  
}

package KZP\_Lab6\_Cholaniuk;  
public class Screwdriver extends Tool {  
 public Screwdriver(String name, String material, int weight) {  
 super(name, material, weight);  
 }  
 @Override  
 public void print() {  
 System.*out*.println("Викрутка: " + getName() + ", Матеріал: " + getMaterial() + ", Вага: " + getWeight() + " г");  
 }  
}

package KZP\_Lab6\_Cholaniuk;  
public class Wrench extends Tool {  
 public Wrench(String name, String material, int weight) {  
 super(name, material, weight);  
 }  
 @Override  
 public void print() {  
 System.*out*.println("Розвідний ключ: " + getName() + ", Матеріал: " + getMaterial() + ", Вага: " + getWeight() + " г");  
 }  
}

