

Міністерство освіти і науки України
Івано-Франківський національний технічний університет нафти і газу

Кафедра КСМ

Лабораторна робота №2
Тема “Створення конструкторів і деструктора”

Виконав студент
групи КІ-18-1
Чуйко Т.В.

Перевірив
Гарасимів Т.Г.

м.Івано-Франківськ
2019р.

Мета:

- вивчити протокол оголошення та опису конструкторів і деструктора;
- закріпити на практиці знання властивостей конструкторів і деструктора;
- набути вміння використовувати конструктори і деструктор при розв'язанні задач.

Умова задачі:

- 1) вивчити теоретичні відомості та методичні вказівки до лабораторної роботи;
- 2) виконати завдання 1 по удосконаленню класу, реалізованого в лабораторній роботі No 1;
- 3) змінити тестуючу програму, реалізовану в лабораторній роботі No 1, для перевірки коректності доповнень, зроблених у класі; відкомпілювати та відладити програму;
- 4) вибрати завдання 2 для виконання згідно з варіантом;
- 5) реалізувати клас та тестуючу програму;
- 6) відкомпілювати та відладити програму;
- 7) відповісти на контрольні запитання;
- 8) зробити висновки.

Загальне завдання 1: доповнити клас, реалізований у лабораторній роботі No 1 конструктором за замовчуванням, конструктором ініціалізації (з самостійно визначеними аргументами) та конструктором копіювання. Перевірити тестуючу програму, послідовно використовуючи кожен з реалізованих конструкторів при створенні об'єкту класу.

Загальне завдання 2: реалізувати та протестувати клас, який містить конструктор за замовчуванням, конструктор копіювання, конструктори ініціалізації згідно з варіантом та деструктор.

6	<p>Статистичний аналізатор файлів</p> <p>Характеризується назвою, довжиною та дескриптором файлу, який аналізується.</p> <p>Підтримує методи:</p> <ul style="list-style-type: none">• визначення наявності у файлі заданого ланцюжка довжиною від 1 до $N / 2$ байтів у файлі розміром N байтів;• визначення кількості повторень заданого ланцюжка довжиною від 1 до $N / 2$ байтів у файлі розміром N байтів. <p>Конструктори:</p> <ul style="list-style-type: none">• з одним аргументом – відкриває файл із заданою назвою;• з двома аргументами – відкриває файл із заданою назвою та запам'ятовує значення його довжини, яке буде використовуватись для аналізу, з другого аргументу.
---	---

Програма мовою Java:

Клас Radio2:

```
public class Radio2 {
    private boolean turnedOn;
    private double frequency;
    private String stationName;
    private int volume;
    private int previousVolume;

    public Radio2() {
        this.turnedOn = false;
        this.frequency = 88;
        this.stationName = null;
        this.volume = 0;
        this.previousVolume = 0;
    }

    public Radio2(boolean turnedOn, double frequency, String stationName, int
volume) {
        this.turnedOn = turnedOn;
        this.frequency = frequency;
        this.stationName = stationName;
        this.volume = volume;
        this.previousVolume = 0;
    }

    public Radio2(Radio2 radio) {
        this.turnedOn = radio.isTurnedOn();
        this.frequency = radio.getFrequency();
        this.stationName = radio.getStationName();
        this.volume = radio.getVolume();
        this.previousVolume = radio.getPreviousVolume();
    }

    public boolean turnOn() {
        this.turnedOn = true;
        this.volume = 5;
        this.frequency = 88;
        return true;
    }

    public boolean turnOff() {
        this.turnedOn = false;
        this.volume = 0;
        return true;
    }

    public boolean increaseVolume() {
        if (volume < 10) {
            this.volume++;
            return true;
        } else {
            return false;
        }
    }
}
```

```

public boolean decreaseVolume() {
    if (volume > 0) {
        this.volume--;
        return true;
    } else {
        return false;
    }
}

public boolean setStation(String stationName, double frequency) {
    if (frequency > 88 && frequency < 108) {
        this.stationName = stationName;
        this.frequency = frequency;
        return true;
    } else {
        return false;
    }
}

public boolean increaseFrequency() {
    if (this.frequency < 108) {
        if (this.volume != 0) {
            this.previousVolume = this.volume;
        }
        this.volume = 0;
        this.stationName = null;
        this.frequency += 0.5;
        return true;
    } else {
        return false;
    }
}

public boolean decreaseFrequency() {
    if (this.frequency > 88) {
        if (this.volume != 0) {
            this.previousVolume = this.volume;
        }
        this.volume = 0;
        this.stationName = null;
        this.frequency -= 0.5;
        return true;
    } else {
        return false;
    }
}

public boolean foundStation(String stationName) {
    this.volume = this.previousVolume;
    this.stationName = stationName;
    System.out.println("Station: " + this.stationName);
    return true;
}

public boolean isTurnedOn() {
    return turnedOn;
}

public double getFrequency() {
    return frequency;
}

public String getStationName() {
    return stationName;
}

```

```

public int getVolume() {
    return volume;
}

public int getPreviousVolume() {
    return previousVolume;
}

@Override
public String toString() {
    return "Radio{" +
        "turnedOn=" + turnedOn +
        ", frequency=" + frequency +
        ", stationName='" + stationName + '\'' +
        ", volume=" + volume +
        ", previousVolume=" + previousVolume +
        '}';
}
}

```

Клас FileAnalyzer:

```

import java.io.File;
import java.io.FileNotFoundException;
import java.util.Scanner;

public class FileAnalyzer {
    private File file;
    private int n;

    public FileAnalyzer() {
        this.file = new File("");
        this.n = 0;
    }

    public FileAnalyzer(String path) {
        this.file = new File(path);
    }

    public FileAnalyzer(String path, int n) {
        this.file = new File(path);
        this.n = n;
    }

    public FileAnalyzer(FileAnalyzer fileAnalyzer) {
        this.file = fileAnalyzer.getFile();
        this.n = fileAnalyzer.getN();
    }

    public boolean hasChain(String chain) {
        try {
            Scanner scanner = new Scanner(file);
            while (scanner.hasNextLine()) {
                if (scanner.nextLine().contains(chain)) {
                    return true;
                }
            }
            scanner.close();
        } catch (FileNotFoundException e) {
            System.out.println("File was not found");
        }
        return false;
    }
}

```

```

public int numberOfChains(String chain) {
    try {
        Scanner scanner = new Scanner(file);
        String fileContent = "";
        while (scanner.hasNextLine()) {
            fileContent += scanner.nextLine();
        }

        int lastIndex = 0;
        int count = 0;
        while (lastIndex != -1) {
            lastIndex = fileContent.indexOf(chain, lastIndex);
            if (lastIndex != -1) {
                count++;
                lastIndex += chain.length();
            }
        }
        scanner.close();

        return count;
    } catch (FileNotFoundException e) {
        System.out.println("File was not found");
    }
    return 0;
}

public File getFile() {
    return file;
}

public int getN() {
    return n;
}
}

```

Клас Main:

```

public class Main {
    public static void main(String[] args) {
        // 1
        Radio2 radio21 = new Radio2();
        System.out.println(radio21.toString());

        Radio2 radio22 = new Radio2(true, 90, "NRJ", 3);
        System.out.println(radio22.toString());

        Radio2 radio23 = new Radio2(radio22);
        System.out.println(radio23.toString());

        // 2
        FileAnalyzer fileAnalyzer = new FileAnalyzer("./lab2.txt");

        System.out.println(fileAnalyzer.hasChain("chain"));
        System.out.println(fileAnalyzer.numberOfChains("chain"));
    }
}

```

Результат:

```
Radio{turnedOn=false, frequency=88.0, stationName='null', volume=0,  
previousVolume=0}  
Radio{turnedOn=true, frequency=90.0, stationName='NRJ', volume=3,  
previousVolume=0}  
Radio{turnedOn=true, frequency=90.0, stationName='NRJ', volume=3,  
previousVolume=0}  
true  
3
```

Висновок: На цій лабораторній роботі я:

- вивчив протокол оголошення та опису конструкторів і деструктора;
- закріпив на практиці знання властивостей конструкторів і деструктора;
- набув вміння використовувати конструктори і деструктор при розв'язанні задач.