

Міністерство освіти і науки, молоді та спорту України
Національний університет “Львівська політехніка”

Кафедра ЕОМ



Звіт

з домашнє завдання №21 з дисципліни:

“Алгоритми та моделі обчислень”

Варіант: № 24.

Виконав:

ст. групи КІ-203

Ширий Богдан Ігорович

Перевірів:

ст. викладач кафедри ЕОМ

Козак Назар Богданович

Львів – 2023

ЗАВДАННЯ:

УМОВА:

Застосовуючи JCL скласти програму (Java), яка за допомогою ***Collections.sort*** дозволяє з вхідного тексту (***String***) вивести слова, що починається з заданої літери (реши слів вивести в алфавітному порядку). Сортювання потрібно виконати без використання додаткової пам'яті, дозволяється використовувати тільки ***ArrayList***, що зберігає індекси на початок слів. Додатково потрібно зберегти результати роботи програми у ***HashMap***.

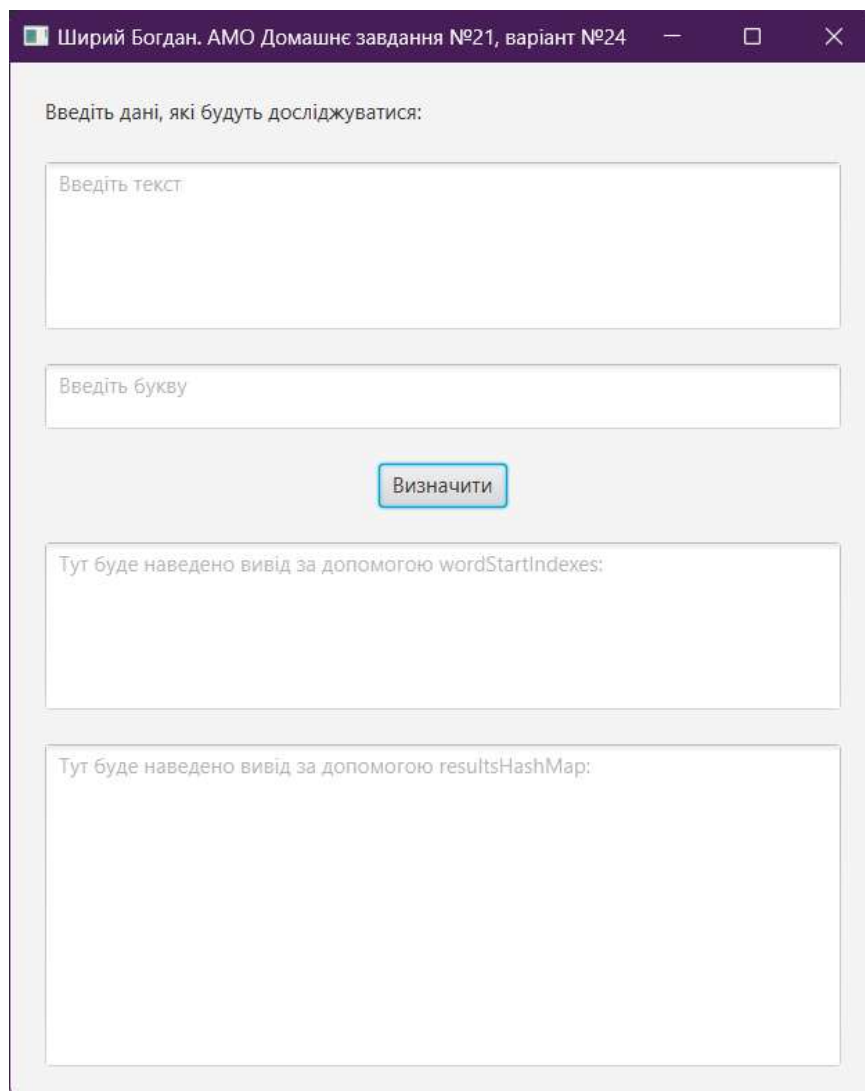
Використати різні способи для виводу результату.

ВИБІР ВАРІАНТУ:

Перша літера мого прізвища латинськими літерами це S.

ВИКОНАННЯ:

Склав Java десктопну програму та її роботу зобразив на рисунках 1 та 2.



Ширий Богдан. АМО Домашнє завдання №21, варіант №24

Введіть дані, які будуть досліджуватися:

Введіть текст

Введіть букву

Визначити

Тут буде наведено вивід за допомогою wordStartIndexes:

Тут буде наведено вивід за допомогою resultsHashMap:

Рис. 1. Початковий вигляд програми.

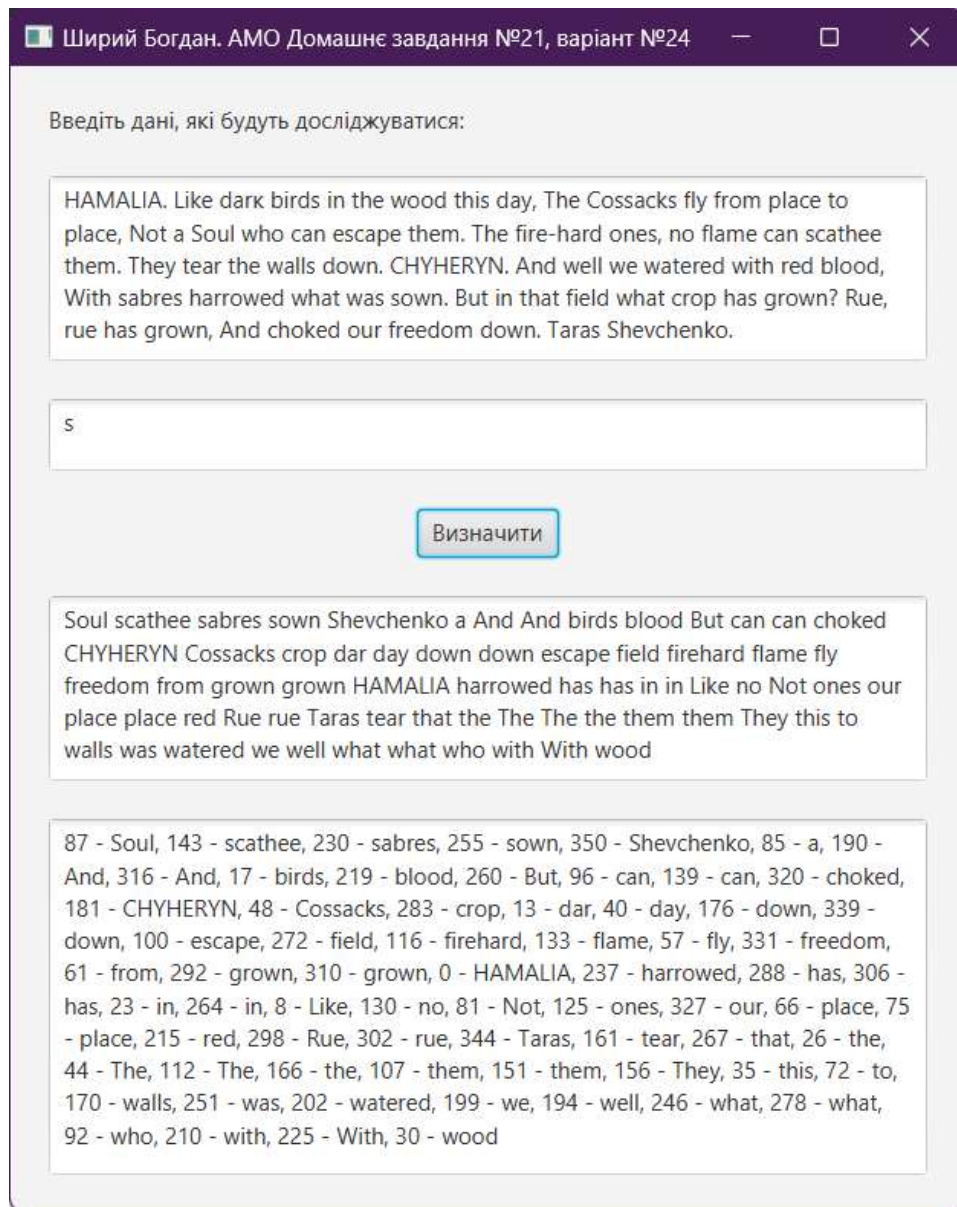


Рис. 2. Програма після натиску на кнопку.

Згідно з вимогами для своєї роботи програма зберігає дані лише в одному String (вхідний текст), ArrayList<Integer> (індекси початку слів) та HashMap<Integer, String> (індекси та їхні слова, результат роботи програми).

Сортування у програмі виконане за допомогою Collections.sort, сама програма виконана за допомогою інструментів бібліотеки JavaFX (Label, TextArea, Button, тощо), у середовищі IntelliJ IDEA Community Edition 2023.1.3.

Відповідно, у лістингах навів код кожного файлу:

- ShyryiApplication.java - лістинг 1;
- acmshyryihw21-view.fxml - лістинг 2;
- ShyryiController.java - лістинг 3.

Лістинг 1. Код файлу ShyryiApplication.java.

```
package com.example.acmshyryihw21;

import javafx.application.Application;
import javafx.fxml.FXMLLoader;
import javafx.scene.Scene;
import javafx.stage.Stage;

import java.io.IOException;

public class ShyryiApplication extends Application {

    @Override
    public void start(Stage stage) throws IOException {
        FXMLLoader fxmlLoader = new
FXMLLoader(ShyryiApplication.class.getResource("acmshyryihw21-view.fxml"));
        Scene scene = new Scene(fxmlLoader.load(), 500, 400);
        stage.setTitle("Ширий Богдан. АМО Домашнє завдання №21, варіант №24");
        stage.setScene(scene);
        stage.show();
    }

    public static void main(String[] args) {
        launch();
    }
}
```

Лістинг 3. Код файлу acmshyryihw21-view.fxml.

```
<?xml version="1.0" encoding="UTF-8"?>

<?import javafx.geometry.Insets?>
<?import javafx.scene.control.Button?>
<?import javafx.scene.control.Label?>
<?import javafx.scene.layout.VBox?>
<?import javafx.scene.control.TextArea?>
<VBox alignment="TOP_LEFT" spacing="20.0"
xmlns:fx="http://javafx.com/fxml"
    fx:controller="com.example.acmshyryihw21.ShyryiController">
    <padding>
        <Insets bottom="20.0" left="20.0" right="20.0" top="20.0"/>
    </padding>
    <Label text="Введіть дані, які будуть досліджуватися:"/>
    <TextArea fx:id="textInputTextArea" promptText="Введіть текст"
wrapText="true" prefHeight="150"/>
    <TextArea fx:id="letterInputTextArea" promptText="Введіть букву"
wrapText="true" prefHeight="20"/>
    <VBox alignment="CENTER">
        <Button text="Визначити" onAction="#onInvestigateButtonClick"/>
    </VBox>
    <TextArea fx:id="textOutputTextArea" promptText="Тут буде наведено
вивід за допомогою wordStartIndexes:" wrapText="true"
prefHeight="150"/>
    <TextArea fx:id="hashMapOutputTextArea" promptText="Тут буде
наведено вивід за допомогою resultsHashMap:" wrapText="true"
prefHeight="240"/>
</VBox>
```

Лістинг 3. Код файлу ShyryiController.java.

```
package com.example.acmshyryihw21;
import javafx.fxml.FXML;
import javafx.scene.control.TextArea;
import java.util.*;
public class ShyryiController {
    @FXML
    private TextArea textInputTextArea;
    @FXML
    private TextArea letterInputTextArea;
    @FXML
    private TextArea textOutputTextArea;
    @FXML
    private TextArea hashMapOutputTextArea;
    @FXML
    protected void onInvestigateButtonClick() {
        String inputText = textInputTextArea.getText()
            .replaceAll("[^a-zA-Z0-9\\s]", "");
        textOutputTextArea.setText("");
        hashMapOutputTextArea.setText("");
        ArrayList<Integer> wordStartIndexes = new ArrayList<>();
        HashMap<Integer, String> resultsHashMap = new LinkedHashMap<>();

        // Знаходимо індекси на початок слів
        for (int i = 0; i < inputText.length(); i++) {
            if (i == 0 && !Character.isWhitespace(inputText.charAt(i))) {
                wordStartIndexes.add(i);
            } else if (i > 0 && !Character.isWhitespace(inputText.charAt(i))
                && Character.isWhitespace(inputText.charAt(i - 1))) {
                wordStartIndexes.add(i);
            }
        }
        // Виводимо слова, що починаються зі шуканої літери
        for (int startIndex : wordStartIndexes) {
            String word = getWordFromIndex(inputText, startIndex);
            if (Character.toLowerCase(word.charAt(0)) == Character
                .toLowerCase(letterInputTextArea.getText().charAt(0))) {
                textOutputTextArea.appendText((word + " "));
                resultsHashMap.put(startIndex, word);
            }
        }
        // Сортуємо решту слів в алфавітному порядку
        Collections.sort(wordStartIndexes, (x, y) -> {
            String wordX = getWordFromIndex(inputText, x);
            String wordY = getWordFromIndex(inputText, y);
            return wordX.compareToIgnoreCase(wordY);
        });
        // Виводимо відсортовані слова
        for (int startIndex : wordStartIndexes) {
            String word = getWordFromIndex(inputText, startIndex);
            if (Character.toLowerCase(word.charAt(0)) != Character
                .toLowerCase(letterInputTextArea.getText().charAt(0))) {
                textOutputTextArea.appendText((word + " "));
                resultsHashMap.put(startIndex, word);
            }
        }
        // Виводимо результати збережені в LinkedHashMap
        for (Map.Entry<Integer, String> entry : resultsHashMap.entrySet()) {
            hashMapOutputTextArea.appendText(entry
                .getKey() + " - " + entry.getValue()
                + (entry.getKey() == wordStartIndexes
                    .get(wordStartIndexes.size() - 1) ? "" : ", "));
        }
    }
    private static String getWordFromIndex(String inputText, int startIndex) {
        int endIndex = inputText.indexOf(' ', startIndex);
        if (endIndex == -1) {
            endIndex = inputText.length();
        }
        return inputText.substring(startIndex, endIndex);
    }
}
```