**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**Чорноморський національний університет   
імені Петра Могили**

**Факультет комп’ютерних наук**

**Кафедра інженерії програмного забезпечення**

**ЗВІТ**

*з лабораторної роботи № 3*

**«Рядки, символи, дата та час»**

**Варіант № 15**

Дисципліна «Об'єктно-орієнтоване програмування»

Спеціальність «Інженерія програмного забезпечення»

121–ЛР.03–209.22210929

***Здобувачка*** *\_\_\_\_\_\_\_\_\_\_****О. К.******Шумакова***

*(підпис)*

*\_\_17.01.2024 \_\_*

*(дата)*

***Викладач*** *\_\_\_\_\_\_\_\_\_\_\_\_\_****Г. В.******Горбань***

*(підпис)*

*\_\_\_\_\_\_\_\_\_\_\_\_\_*

*(дата)*

**Миколаїв – 2024**

**Лабораторна робота №3**

**Тема роботи:**Рядки, символи, дата та час.

**Завдання:**

У методі main() створити меню за допомогою конструкції switch для вводу даних, виклику створених методів та виходу з програми.

**Реалізувати наступні методи:**

1. Метод, який приймає рядок та роздруковує всі можливі сполучення символів, з яких складається цей рядок. Наприклад, для вхідного рядку "GOD" можливі 6 перестановок, отже в результаті мають бути роздруковані наступні рядки: "GOD", "OGD", "DOG", "GDO", "ODG", "DGO".
2. Метод, який перевіряє, чи є вхідний рядок паліндромом (однаково читається у любому напрямку, наприклад radar). Не використовувати методи класу *StringBuilder*.
3. Використовуючи класи з пакету java.time, вивести на консоль інформацію про дату та час польоту з одного часового поясу (пункт А) в інший (пункт Б). Задати дату/час вильоту в поточному часі та тривалість польоту (кількість годин та хвилин). Розрахувати та вивести на консоль:
   * дату/час вильоту та прильоту в часовому поясі пункту А;
   * дату/час вильоту та прильоту в часовому поясі пункту Б;
   * дату/час вильоту та прильоту в форматі UTC.

**Лістинг коду:**

**import** java**.**time**.\*;**

**import** java**.**time**.**format**.**DateTimeFormatter**;**

**import** java**.**util**.**Scanner**;**

public class Main **{**

public static void main**(**String**[]** args**)** **{**

Scanner scanner **=** **new** Scanner**(**System**.**in**);**

**while** **(true)** **{**

System**.**out**.**println**(**"1. Output the permutations"**);**

System**.**out**.**println**(**"2. Check for palindrome"**);**

System**.**out**.**println**(**"3. Output fly time"**);**

System**.**out**.**println**(**"4. Exit"**);**

System**.**out**.**print**(**"Input options number: "**);**

String option **=** scanner**.**nextLine**();**

**switch** **(**option**)** **{**

**case** "1"**:**

System**.**out**.**print**(**"Input str: "**);**

String input **=** scanner**.**nextLine**();**

printPermutations**(**input**,** ""**);**

**break;**

**case** "2"**:**

System**.**out**.**print**(**"Input str: "**);**

String str **=** scanner**.**nextLine**();**

System**.**out**.**println**(**

isPalindrome**(**str**)**

**?** "True: this is Palindrome"

**:** "False: This is not a palindrome"

**);**

**break;**

**case** "3"**:**

System**.**out**.**print**(**"Введіть пункт відправлення: "**);**

String departureLocation **=** scanner**.**nextLine**();**

System**.**out**.**print**(**"Введіть пункт призначення: "**);**

String arrivalLocation **=** scanner**.**nextLine**();**

System**.**out**.**print**(**"Введіть дату вильоту (формат: yyyy-MM-dd): "**);**

String departureDate **=** scanner**.**nextLine**();**

System**.**out**.**print**(**"Введіть час вильоту (формат: HH:mm): "**);**

String departureTime **=** scanner**.**nextLine**();**

System**.**out**.**print**(**"Введіть тривалість польоту в годинах: "**);**

int hours **=** scanner**.**nextInt**();**

System**.**out**.**print**(**"Введіть тривалість польоту в хвилинах: "**);**

int minutes **=** scanner**.**nextInt**();**

scanner**.**nextLine**();**

System**.**out**.**print**(**

"Введіть часовий пояс пункту відправлення (наприклад, Europe/Kiev): "

**);**

String departureZoneId **=** scanner**.**nextLine**();**

System**.**out**.**print**(**

"Введіть часовий пояс пункту призначення (наприклад, America/New\_York): "

**);**

String arrivalZoneId **=** scanner**.**nextLine**();**

printFlightTimes**(**

departureLocation**,**

arrivalLocation**,**

departureDate**,**

departureTime**,**

hours**,**

minutes**,**

departureZoneId**,**

arrivalZoneId

**);**

**break;**

**case** "4"**:**

**return;**

**default:**

System**.**out**.**println**(**"Невідома опція"**);**

**}**

**}**

**}**

private static void printPermutations**(**String str**,** String ans**)** **{**

**if** **(**str**.**length**()** **==** 0**)** **{**

System**.**out**.**println**(**ans**);**

**return;**

**}**

**for** **(**int i **=** 0**;** i **<** str**.**length**();** i**++)** **{**

char ch **=** str**.**charAt**(**i**);**

String ros **=** str**.**substring**(**0**,** i**)** **+** str**.**substring**(**i **+** 1**);**

printPermutations**(**ros**,** ans **+** ch**);**

**}**

**}**

private static boolean isPalindrome**(**String str**)** **{**

int i **=** 0**,** j **=** str**.**length**()** **-** 1**;**

**while** **(**i **<** j**)** **{**

**if** **(**str**.**charAt**(**i**)** **!=** str**.**charAt**(**j**))** **return** **false;**

i**++;**

j**--;**

**}**

**return** **true;**

**}**

private static void printFlightTimes**(**

String departureLocation**,**

String arrivalLocation**,**

String departureDate**,**

String departureTime**,**

int hours**,**

int minutes**,**

String departureZoneId**,**

String arrivalZoneId

**)** **{**

DateTimeFormatter dateFormatter **=** DateTimeFormatter**.**ofPattern**(**"yyyy-MM-dd"**);**

DateTimeFormatter timeFormatter **=** DateTimeFormatter**.**ofPattern**(**"HH:mm"**);**

LocalDate localDate **=** LocalDate**.**parse**(**departureDate**,** dateFormatter**);**

LocalTime localTime **=** LocalTime**.**parse**(**departureTime**,** timeFormatter**);**

ZonedDateTime departureDateTime **=** ZonedDateTime**.**of**(**

localDate**,**

localTime**,**

ZoneId**.**of**(**departureZoneId**)**

**);**

ZonedDateTime arrivalDateTime **=** departureDateTime

**.**plusHours**(**hours**)**

**.**plusMinutes**(**minutes**)**

**.**withZoneSameInstant**(**ZoneId**.**of**(**arrivalZoneId**));**

DateTimeFormatter formatter **=** DateTimeFormatter**.**ofPattern**(**

"dd.MM.yyyy HH:mm:ss VV"

**);**

System**.**out**.**println**(**

"Час вильоту з " **+**

departureLocation **+**

" (місцевий): " **+**

formatter**.**format**(**departureDateTime**)**

**);**

System**.**out**.**println**(**

"Час прильоту в " **+**

arrivalLocation **+**

" (місцевий): " **+**

formatter**.**format**(**arrivalDateTime**)**

**);**

System**.**out**.**println**(**

"Час вильоту з " **+**

departureLocation **+**

" (UTC): " **+**

formatter**.**format**(**departureDateTime**.**withZoneSameInstant**(**ZoneOffset**.**UTC**))**

**);**

System**.**out**.**println**(**

"Час прильоту в " **+**

arrivalLocation **+**

" (UTC): " **+**

formatter**.**format**(**arrivalDateTime**.**withZoneSameInstant**(**ZoneOffset**.**UTC**))**

**);**

**}**

**}**

**Результати роботи програми:**

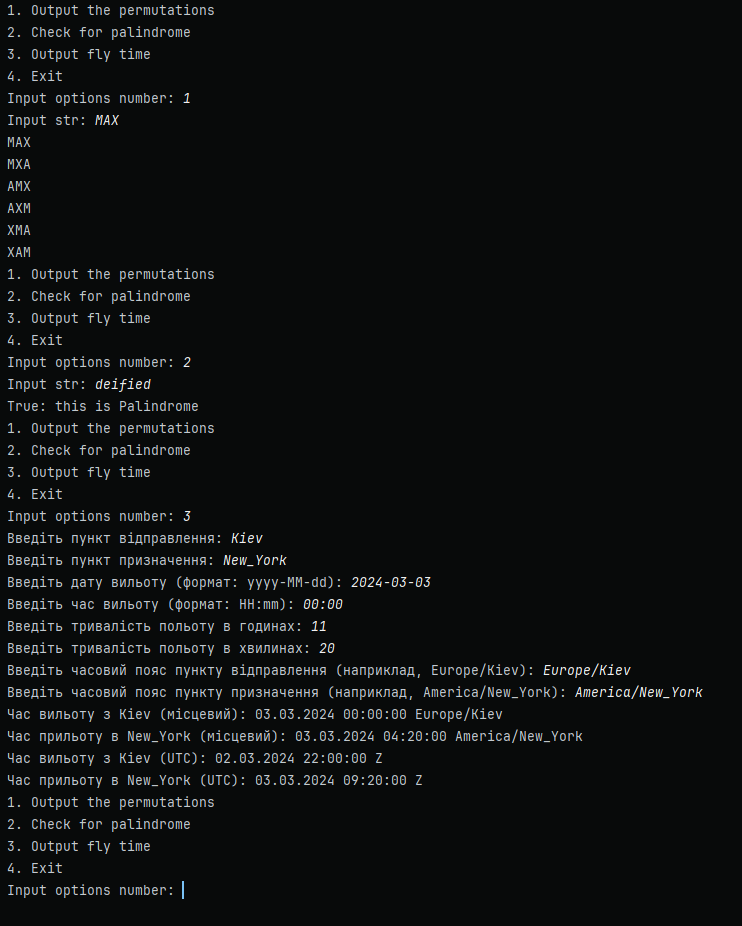


Рисунок 1