Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра ЕОМ



до лабораторної роботи N 2

з дисципліни: «Кросплатформні засоби програмування» «Дослідження базових конструкцій мови Java» Варіант №22

Виконав:

ст.гр. КІ-34

Трач В.І.

Прийняв:

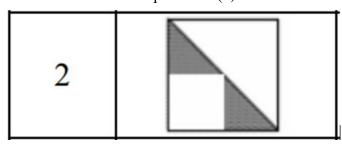
Іванов Ю.С.

Мета роботи: ознайомитися з базовими конструкціями мови Java та оволодіти навиками написання й автоматичного документування простих консольних програм мовою Java.

Завдання:

- 1. Написати та налагодити програму на мові Java згідно варіанту. Програма має задовольняти наступним вимогам:
 - програма має розміщуватися в загальнодоступному класі Lab2ПрізвищеГрупа;
 - програма має генерувати зубчатий масив, який міститиме лише заштриховані області квадратної матриці згідно варіанту;
 - розмір квадратної матриці і символ-заповнювач масиву вводяться з клавіатури;
 - при не введені або введенні кількох символів-заповнювачів відбувається коректне переривання роботи програми;
 - сформований масив вивести на екран і у текстовий файл;
 - програма має володіти коментарями, які дозволять автоматично згенерувати документацію до розробленої програми.
- 2. Автоматично згенерувати документацію до розробленої програми.
- **3.** Скласти звіт про виконану роботу з приведенням тексту програми, результату її виконання та фрагменту згенерованої документації.
- 4. Дати відповідь на контрольні запитання.

Варіант: 22(2)



Теоретичний матеріал:

Автоматичне документування При автоматичній генерації документації використовується утиліта javadoc, яка аналізує вміст між / і / та на його базі генерує документацію у форматі.html. Коментарі між / і / прийнято починати з описового тексту, за яким слідують дескриптори. Використання дескрипторів полегшує як автоматичну генерацію документації, так і розуміння коду, до якого відноситься коментар. Дескриптор, на відміну від решти коментарів, починається з символу @ за яким слідує ім'я дескриптора. Оскільки документація генерується у форматі.html, то між / і */ допускається розташування html-тегів, включаючи рисунки. Для автоматичної генерації документації між / і / можна розмістити:

- коментарі до класу;
- коментарі до методів;
- коментарі до полів;
- загальні коментарі.

Коментарі до класу мають бути розміщені після директив ітрог безпосередньо перед визначенням класу. Найчастіше цей коментар має вигляд одного або кількох коротких речень: /** Об'єкт класу Person описує особу. Особа має властивості: ім'я, прізвище та стать./ Коментарі до методів розташовуються безпосередньо перед методами, які вони описують. Крім дескрипторів загального призначення для коментування методів використовуються дескриптори:

- @рагат змінна опис Цей дескриптор додає в опис методу розділ "рагатеters". Опис цього елементу може складатися з кількох рядків та містити html-теги. Всі дескриптори @рагат, що відносяться до одного методу слід групувати разом.
- @return опис Цей дескриптор додає в опис методу розділ "returns". Опис цього елементу може складатися з кількох рядків та містити html-теги.
- @throws опис_класу

Код програми:

```
pw.println("Enter size of matrix : " + size);
pw.println("Enter symbol-filler : " + symbol);
System.out.print("Error! : " + e);
```

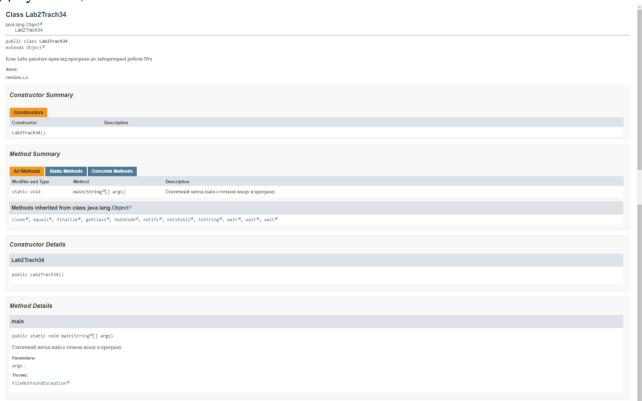
}

Результат виконання програми:

Вивід в консоль:

Вивід в файл:

Документація



Висновок: Я ознайомився з базовими конструкціями мови Java та оволодів навиками написання й автоматичного документування простих консольних програм мовою Java.