Міністерство освіти і науки України

Національний університет «Львівська Політехніка»

Інститут комп’ютерних наук та інформаційних технологій

**ЛАБОРАТОРНА РОБОТА№1**

З дисципліни **«**Прикладне програмування**»**

Тема:

**«Основи Java»**

|  |
| --- |
| **Виконав:** |
| **Грева Олег Олегович** |
| **Група:**  **КН-204** |

|  |
| --- |
| **Прийняв:** |
| **Вергун В.Р.** |
|  |

Львів 2020

**Індивідуальне завдання**

1.Напишіть HelloWorld програму, яка виводить на екран довільне

повідомлення. Скомпілюйте та запустіть її через командний

рядок.

2. Створіть клас, що містить int та char, які не є ініціалізованими.

Виведіть на екран їхні значення щоб перевірити, що Java здійснює

ініціалізацію за замовчуванням.

3. Напишіть програму, яка виводить на екран три аргументи, які

передаються через командний рядок.

4. Напишіть програму (з використанням ООП підходу), що

відповідає наступним вимогам:

• Користувач вводить інтервал (наприклад, [1;12]).

• Програма виводить на екран непарні числа з інтервалу за

зростанням і парні числа за спаданням.

• Програма виводить на екран суму непарних та парних чисел.

• Програма будує ряд Фібоначчі: першим числом буде найбільше

непарне число, другим – найбільше парне число. Довжину

ряду користувач вводить з клавіатури.

• Програма виводить відсоток непарних і парних чисел

Фібоначчі.

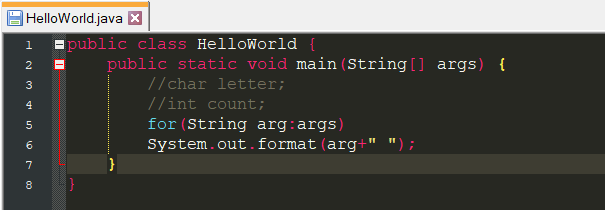
5. Продокументуйте програму з п.4 з використанням JavaDoc і

згенеруйте на основі нього документацію.

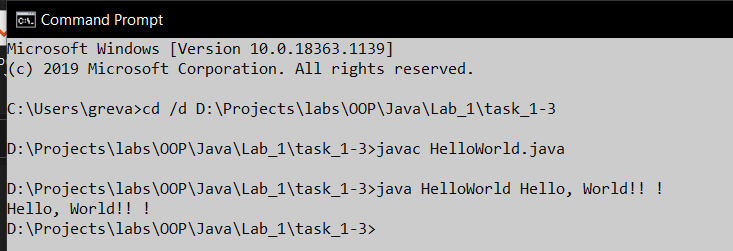
**Хід виконання роботи**

1. Компіляція програми через командний рядок :

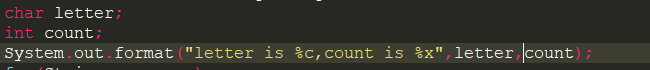
Створюю текстовий файл з розширенням .java ,в якому записую поданий код.



Компілюю файл за допомогою вбудованого в JDK компілятора , потім запускаю проект , використовуючи інструмент з JDK.



2. При виводі неініціалізованих змінних виводяться значення за замовчуванням, що свідчить про наявність ініціалізації за замовчуванням.



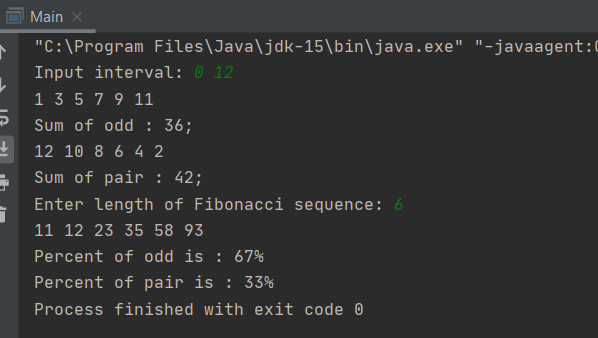
3. Програма з пункту 1 приймає довільне число параметрів у вигляді рядків та виводить їх один за одним.

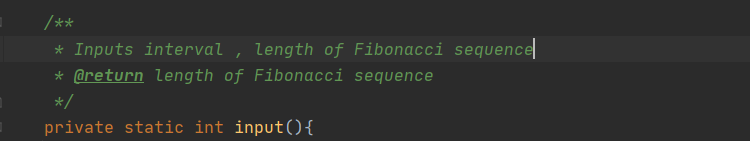
**Лістинг програми**

import java.util.Scanner;  
  
public class Main {  
 final private static Scanner *input* = new Scanner(System.*in*);  
 private static int *start*;  
 private static int *end*;  
  
 public static void main(String[] args) {  
 Main.*input*();  
 Main.*fibonacci*(Main.*odd*(),Main.*pair*());  
 }  
  
 private static void input(){  
 System.*out*.print("Input interval: ");  
 Main.*start* = *input*.nextInt();  
 Main.*end* = *input*.nextInt();  
 }

private static int pair(){  
 int sum = 0;  
 for(int num = Main.*end*;num != Main.*start*;num--) {  
 if(num % 2 == 0){  
 System.*out*.print(num + " ");  
 sum+=num;  
 }  
 }  
 System.*out*.print("\nSum of pair : "+sum+";\n");  
 return Main.*end* % 2 == 0 ?Main.*end*:Main.*end* - 1;  
 }  
  
 private static int odd(){  
 int sum = 0;  
 for(int num = Main.*start*;num != Main.*end*;num++) {  
 if(num % 2 != 0){  
 System.*out*.print(num + " ");  
 sum += num;  
 }  
 }  
 System.*out*.print("\nSum of odd : "+sum+ ";\n");  
 return Main.*end* % 2 == 0 ?Main.*end* - 1:Main.*end*;  
 }  
 private static void fibonacci(int first,int second){  
 System.*out*.print("Enter length of Fibonacci sequence: ");  
 int len = *input*.nextInt();  
 int prev = first,next = second,pairP = 0, oddP, num = len;  
  
 for(len/=2;len != 0;len--){  
  
 if(prev%2==0)  
 pairP++;  
 if(next%2==0)  
 pairP++;  
  
 System.*out*.format("%d %d ",prev,next);  
  
 prev += next;  
 next += prev;  
 }  
 pairP = (pairP\*100/num);  
 oddP = 100 - pairP;  
 System.*out*.print("\nPercent of odd is : "+oddP+"%\n");  
 System.*out*.print("Percent of pair is : "+pairP+"% ");  
 }  
}

**Результат роботи програми**



5. Використовуючи спеціальний синтаксис javadoc формую коментарі до методів та класу. 

За допомогою вбудованого функціоналу генерується документація на основі створених коментарів.

**Висновок**

Зрозумів процес створення програм на мові програмування Java , навчився генерувати документацію використовуючи JavaDoc.