Object-Oriented Programming: Homework #2

Time

- Assigned on 3/13/2023 (Monday); Due at 09:00 on 3/27/2023 (Monday)

Submission

- Source Code Submit on iLearning.
- Your program code for this homework should be submitted as a single .java file named HW2_studentID.java (**DO NOT** submit the entire Eclipse project, only the source files).
- In addition, please delete the package before uploading the file.

Objectives

- Another example of the use of Java input and output streams.
- Familiarity with selections, iterations, and simple functions.
- Importance of designing interesting test cases.

Program Descriptions

- In a distant country, a queen is very superstitious and does not like numbers with repeated digits like 77, 434 or 2121. Thus, she declares any house street number with repeated digits to be invalid. You have been appointed by Her Majesty to write a program that, given two integers $\boldsymbol{a}$ and $\boldsymbol{b}$, determines the maximum number of valid house street numbers between $\boldsymbol{a}$ and $\boldsymbol{b}$, inclusive.

- Examples:
  - For $\boldsymbol{a} = \mathbf{3}$ and $\boldsymbol{b} = \mathbf{12}$, the answer is 9.
  - For $\boldsymbol{a} = \mathbf{12}$ and $\boldsymbol{b} = \mathbf{34}$, the answer is 21.

Program Specifications

- When developing your solution to this problem, ensure that your program conforms to the following requirements and assumptions:
    1. Implementation is written such that it is readable by other programmers. Use descriptive variable identifiers and comments where appropriate (non-trivial program code).
    2. You cannot use **string** variables in this assignment.
    3. The input/output format should be exactly as follows. The Input is given by two integers a and b such that $0 < a \leq b \leq 10000$.
    4. Your code should check if the input is given as specified: the numbers are positive integers, the second one is greater than or equal to the first, and so on. If the input does not meet the requirements, the output should be "Invalid input".
    5. Your program must define and use the following function:

       ```
       public static int countDigitOccurrence (int n, int digit)
       ```

       This function has as input parameters a positive integer n that is in the range and $0 \leq digit \leq 9$, and returns how many times the digit occurs in the number n.

Example Program Run

- Required I/O format (user input in **red**; everything else is output):

```
Enter numbers a <= b: 12 34↵
There are 21 valid numbers between 12 and 34↵

Enter numbers a <= b: -7 10↵
Invalid input↵

Enter numbers a <= b: 12 100230↵
Invalid input↵

Enter numbers a <= b: 121 62↵
Invalid input↵

Enter numbers a <= b: 1 5↵
There are 5 valid numbers between 1 and 5↵

Enter numbers a <= b: 121 122↵
There are 0 valid numbers between 121 and 122
```