**CSCI 2302**

**Object-Oriented Thinking Chapter**

**regex Lab**

Contents

[Intro 1](#_Toc95048531)

[Learning Goals 1](#_Toc95048532)

[Notes 1](#_Toc95048533)

[Task 1](#_Toc95048534)

[Submit 2](#_Toc95048535)

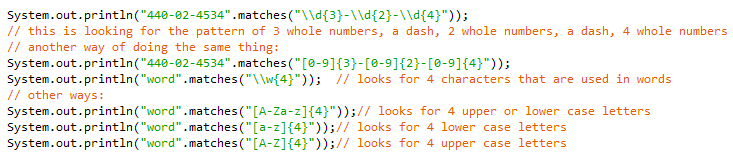
Intro: Regular expressions, or regex for short, are one of the most powerful and applicable techniques in programming. We can use regular expressions to search for and find patterns in strings. They can be used in nearly every language, and they allow us to validate user input, perform search tasks, and even test code. Learn the rules of regular expressions so that you can apply them as needed into your programs, adding the robustness that is needed when dealing with user input.

Learning Goals: To acquire the skill of creating patterns, to apply that skill to several problems, to relate this skill for future problems.

Notes: please see the Java regex Notes.docx file.

Task: Complete the steps outlined below in a file named YourmysfaUsername\_regexLab.java.

1. Validate user input for a phone number.
   1. Ask the user to enter a phone number in a given format (e.g. ###-###-#### or ###.###.#### or ##########).
   2. Validate and state to the screen if the input is a valid phone number.



Sample run:

Please enter your phone number:

936-468-2508

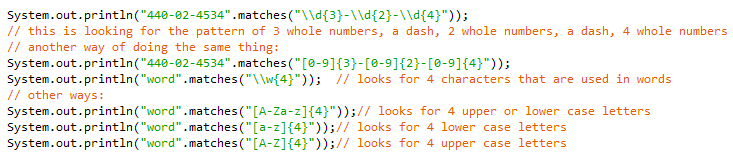
936-468-2508 is a valid phone number format

Please enter your phone number:

9369468-2508

9369468-2508 is not a valid phone number format

1. Validate license plates.
   1. Input: regexlab.txt; output to the screen
   2. A valid license plate has the format of three upper case letters or digits, a space, and then four upper case letters or digits. Based on this format, write a method that will validate proposed license plate data that is stored on the file, regexlab.txt.
      1. Read in a license plate from the file (until there is no more to be read).
      2. Use the method for regex with the format and state (print to the screen) if an entry is valid or invalid for a license plate.

****

Example of input: from file regexlab.dat

ABC 1234

Example of output:

The proposed license plate is ABC 1234.

It is a valid license plate.

Submit: Submit your YourmysfaUsername\_regexLab.java file in the Dropbox in Brightspace by D2L.