

Objectives:

- 1. To program common operations for two-dimensional arrays
- 2. Create Classes and Objects.
- 3. Process Array of objects.
- 4. To use UML graphical notation to describe classes and objects.
- 5. To use the **String** class to process immutable strings.
- 6. To use the **StringBuilder** class to process mutable strings.

Specification

Submission: Online through Ritaj.

What to submit: Your OWN well-structured and well-commented JAVA files (.java) + PDF file (compressed

into a studentId_lab#.rar file, e.g. 1234567_lab1.rar).

Deadline: 22/3/2019 by midnight. (The online submission will be disabled after this time).

Important

- The assignment solutions are found in the internet.
- Academic honesty:
 - o This is an individual assignment. Individual assignments must be each student's own work.
 - Copying 1 line from a friend or the internet will be considered cheating.
 - Cheating will result in an official university disciplinary review and the University regulations will be strictly enforced.

Tasks

Task 1: Algebra: 2×2 matrix inverse

The inverse of a square matrix A is denoted A^{-1} , such that $A \times A^{-1} = I$, where I is the identity matrix with

all 1s on the diagonal and 0 on all other cells. For example, the inverse of matrix $\begin{bmatrix} 3 & 4 \end{bmatrix}$ is $\begin{bmatrix} 1.5 & 0 \end{bmatrix}$, i.e.,

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \times \begin{bmatrix} -0.5 & 1 \\ 1.5 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

The inverse of a 2×2 matrix A can be obtained using the following formula:

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \qquad A^{-1} = \frac{1}{ad - bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

Implement the following method to obtain an inverse of the matrix:

public static double[][] inverse(double[][] A)

The method returns **null** if ad – bc is 0.

Write a test program that prompts the user to enter **a**, **b**, **c**, **d** for a matrix, and displays its inverse matrix. Here is a sample run:

Sample run 1:	
	Enter a, b, c, d: 1234 🖑
	-2.0 1.0
	1.5 -0.5
Sample run 2:	
	Enter a, b, c, d: 0.5 2 1.5 4.5 🖋
	-6.0 2.66666666666665
	2.0 -0.66666666666666
Sample run 3:	
	Enter a, b, c, d: 1 2 3 6 4
	No inverse matrix

Task 2: Bioinformatics: find genes

Biologists use a sequence of the letters **A**, **C**, **T**, and **G** to model a **genome**. A **gene** is a substring of a genome that starts after a triplet **ATG** and ends before a triplet **TAG**, **TAA**, or **TGA**. Furthermore, the length of a gene string is a multiple of **3**, and the gene does not contain any of the triplets **ATG**, **TAG**, **TAA**, or **TGA**.

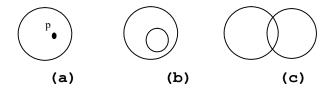
Write a program that prompts the user to enter a genome and displays all genes in the genome. If no gene is found in the input sequence, display "no gene is found". Here are the sample runs:

```
Enter a genome string: TTATGTTTTAAGGATGGGCGTTAGTT 
TTT
GGGCGT
Sample run 2:
Enter a genome string: TGTGTGTATAT 
no gene is found
```

Task 3: Circle2D class

Define the Circle2D class that contains:

- Two double data fields named \underline{x} and \underline{y} that specify the center of the circle with get methods.
- A data field <u>radius</u> with a <u>get</u> method.
- A no-arg constructor that creates a default circle with (0, 0) for (x, y) and 1 for radius.
- A constructor that creates a circle with the specified <u>x</u>, <u>y</u>, and <u>radius</u>.
- A method getArea() that returns the area of the circle.
- A method getPerimeter() that returns the perimeter of the circle.
- A method <u>contains(double x, double y)</u> that returns <u>true</u> if the specified point $(\underline{x}, \underline{y})$ is inside this circle. See the following figure (a).
- A method <u>contains(Circle2D circle)</u> that returns <u>true</u> if the specified circle is inside this circle. See the following figure (b).
- A method <u>overlaps(Circle2D circle)</u> that returns <u>true</u> if the specified circle overlaps with this circle. See the following figure (c).



- (a) A point is inside the circle.
- (b) A circle is inside another circle.
- (c) A circle overlaps another circle.
- 1. Draw the UML diagram for the class. (use any free online tool to draw UML and save it as a PDF file. e.g. https://www.draw.io/)
- 2. Implement the class.
- 3. Write a test program that creates a <u>Circle2D</u> object <u>c1</u> (<u>new Circle2D(2, 2, 5.5)</u>), displays its area and perimeter, and displays the result of <u>c1.contains(3, 3)</u>, <u>c1.contains(new Circle2D(4, 5, 10.5))</u>, and <u>c1.overlaps(new Circle2D(3, 5, 2.3))</u>.

Good Luck!