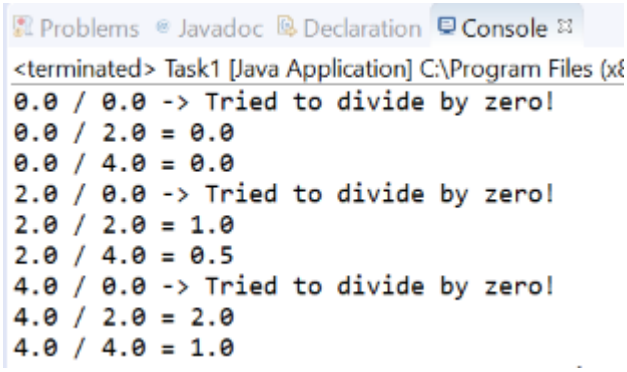
OPMS 2018 - Exercise 4

Task 1 (Exceptions)

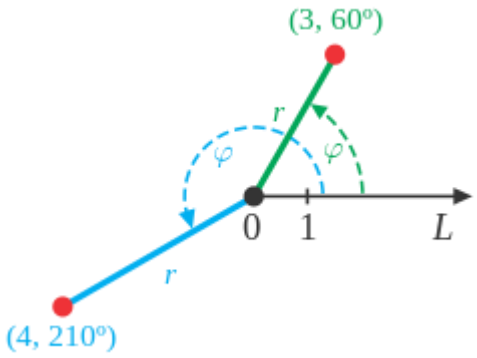
1. Create a new project called “Exercise4” and create a new class called `MathFunctions` which has a main()-method.
2. Create a static function `division`, which takes two double arguments and returns the division.
3. As you might have already noticed, not all input arguments will return a valid result. Since a division by 0 is impossible, this case has to be handled by throwing an exception. Implement a new class `DivisionByZeroException` which extends Exception.
4. Throw the DivisionByZeroException in your `division` function if the second argument is 0.
5. Test your implementation by calling your function from the main()-method. Use a try-catch block to print the result of the division, if the parameters are valid and else print an error message. See Picture 1 for a possible output.



Picture 1: Possible output from testing the division function with different parameters

Task 2 (Exceptions)

The polar coordinate system describes each point in a plane by a distance and an angle.



Picture 2: The arctan function

1. Extend your class `MathFunctions` by a static function `getPolarAngle` which takes the coordinates x and y as two double arguments. The function should implement the behavior

$$\theta = \begin{cases} \arctan(\frac{y}{x}) & \text{if } x > 0 \\ \arctan(\frac{y}{x}) + \pi & \text{if } x < 0 \text{ and } y \geq 0 \\ \arctan(\frac{y}{x}) - \pi & \text{if } x < 0 \text{ and } y < 0 \\ \frac{\pi}{2} & \text{if } x = 0 \text{ and } y > 0 \\ -\frac{\pi}{2} & \text{if } x = 0 \text{ and } y < 0 \\ \text{undefined} & \text{if } x = 0 \text{ and } y = 0 \end{cases}$$

Picture 3: Calculation of a polar angle from Cartesian coordinates

   Throw an IllegalArgumentException if x and y are both 0. Use `Math.atan` for calculating arctan() and `Math.PI` for π.
2. Provide the IllegalArgumentException with the message String "x and y cannot both be 0 at the same time.". Test your function in the main()-method and print the exception message in the catch block by using the exception's getMessage()-method.

Task 3 (Data Structures)

In this task, you will use a list to save student names and work with them.

1. Create a new class `Task3`. At the top, add `import java.util.ArrayList;` to the top to import lists.
2. Create a main method. In this main method, write your code.
3. Create a variable `ArrayList<String> students = new ArrayList<>()`
4. Add your name to the list.
5. Add the names of 2 of your neighbors to the list.
6. Use `System.out.println(students.toString());` to see what is in your list.

   What is the result?
   Answer: _____

7. Add "WALL-E" to the list at index 1.
   Hint: Use `students.add(index, element)` to add an element to a specific index.

   Use the command from 5. to print the list. What is the result now?
   Answer: _____

8. Remove your name from the list.
   Hint: Use `students.remove(element)` to do so.

   Print the list again. What is the result now?
   Answer: _____

9. Clear the list and print it again.

   What is the command to remove all elements from a list?
   Answer: _____

   What does your print command print?
   Answer: _____