

RDR_Rubber Duck Race JUnit 5 & TDD

In US they invented Rubber Duck Races. A lot of YouTube videos shows it, f.ex.

https://www.youtube.com/watch?v=NIUm_kBCW9g



The Americans throw a large amount of numbered yellow rubber ducks in streaming water.

The owner of the first duck, which crosses the finishing line, wins very often a major prize. You pay to participate and it is possible to play upon the ducks too. The money usually goes to charity.

Technical HINTs

Divide and Conquer.

Take the problem and rephrase it as smaller problems. Give this smaller problems a number and/or name. It is much easier to solve the small problems one by one, than one big one. Especially, when we are getting involved in bigger projects.

Test Driven Development (TDD)

Your program must be developed by Test First technology. That is by making Junit 5 test before you code. The faster this becomes a good habit, the easier it is to use.

Use queues in the solution.

And what is a queue?

As Reges writes :

Collection

An object which stores a group of other objects, called its elements

A Queue is a collection with FiFo, First In First Out. Just as a row before a ticket office. The first one to enter the queue should be the first one to get a ticket.

Take the exercise parts below, one by one. Define each of them as smaller tasks (document it), and solve each of them. Remember to use TDD / Junit 5 and Queues in the final solution.

Upload document and code in a zip-file to Fronter.

Exercise 1

Construct your own Queue class. It must be able to add a new object in the end and remove the oldest one from the front of the queue.

It must be possible too to get the number of objects in the queue.

The Queue class can be based upon an ArrayList.

Yes Java has (of course) a queue class. But do still try to build your own adding the services you want it to have too.

Exercise 2

We simulate in this exercise a Rubber Duck Race with for – each time step – a number N of queues. They do each have room for up to same number N of unique numbered yellow rubber ducks.

We begin with 10 queues of ten yellow rubber ducks, i.e. there are 100 numbered ducks (1 – 100) in our race.

For each time step a random number from 1 – N are drawn, and the foremost rubber duck from that queue is transferred via a new random number to the next set of queues.

But for each time step in our simulation, there 1 less queue with one less room. I.e. in time step number 2, there are only room for the first 81 ducks from time step number 1.

In time step number 10, there will only be one duck – our winner.

Implement this and show the queues for every time step.

Exercise 3

Set up the odds for in which position in its queue the winning duck will start from.

