Exercise 1

This exercise might be done during the Tutorium

Write a class with two `public` methods, returning the discrepancy and the star discrepancy, respectively, of a set $\{x_1, \dots, x_n\}$ of one-dimensional points. These points are not supposed to be sorted when given. Test your implementation computing the discrepancy and the star discrepancy of the sets of Exercise 3 Handout 4, i.e.,

$$A_1 = \{1/8, 1/4, 1/2, 3/4\}$$

and

$$A_2 = \{1/4, 1/2, 5/8, 3/4\}.$$

As seen in the exercise solution, you have to get

$$D(A_1) = \frac{3}{8}, \quad D^*(A_1) = \frac{1}{4}, \quad D(A_2) = \frac{1}{2}, \quad D^*(A_2) = \frac{1}{4}.$$

Hint:

The discrepancy may be computed as

$$D(\{x_1, \dots, x_n\}) = \max_{a \in \{0, x_1, \dots, x_n\}} \max_{b \in \{x_1, \dots, x_n, 1\}, b > a} \max\left(b - a - \frac{|x_i \in (a,b)|}{n}, \frac{|x_i \in [a,b]|}{n} - (b-a)\right). \quad (1)$$

One can then use representation (1) by first computing

$$\max_{b \in \{x_1, \dots, x_n, 1\}, b > a} \max\left(b - a - \frac{|x_i \in (a,b)|}{n}, \frac{|x_i \in [a,b]|}{n} - (b-a)\right) \quad (2)$$

for $a \in \{x_1, \dots, x_n\}$ fixed, and then computing the discrepancy as the maximum between the star discrepancy, which is

$$\max_{b \in \{x_1, \dots, x_n, 1\}, b > a} \max\left(b - \frac{|x_i \in (0,b)|}{n}, \frac{|x_i \in [0,b]|}{n} - b\right),$$

and the maximum of the values of (2).

Exercise 2

This exercise might be done during the Tutorium

Implement the methods

```
getVanDerCorputStarDiscrepancy(int sequenceLength, int base)
```

and

```
getVanDerCorputDiscrepancy(int sequenceLength, int base)
```

in the class `VanDerCorputDiscrepancy` that you find in `com.andreamazzon.exercise5.discrepancy`. Here you have to compute the star discrepancy and the discrepancy, respectively, of a Van der Corput sequence of given length and base.

Once we are able to compute the discrepancy of Van der Corput sequences with the methods above, the methods

```
plotVanDerCorputStarDiscrepancy(int maxSequenceLength, int base)
```

and

```
plotVanDerCorputDiscrepancy(int maxSequenceLength, int base)
```

plot the star discrepancy and the discrepancy of Van der Corput sequences for increasing dimensions. Here you only have to complete the definition of the `DoubleUnaryOperator starDiscrepancyFunction` and `discrepancyFunction`, respectively.

Call these methods in a class you can run in order to observe the plots.

Exercise 3

Consider an interface `UniformRandomNumberSequence` taking care of the generation of a sequence of (pseudo) random numbers uniformly distributed in the interval $(0, 1)$. Suppose that this interface has a method

```
double[] getSequenceOfRandomNumbers(),
```

which returns a one-dimensional array of random numbers uniformly distributed in $(0, 1)$.

Imagine now you write a class `TwoDimensionalFunctionIntegration` whose goal is to compute the Monte-Carlo approximation of an integral

$$\int_0^1 \int_0^1 f(x, y) dx dy, \tag{3}$$

where $f : (0, 1) \times (0, 1) \to \mathbb{R}$. Assume this class has a constructor

```
TwoDimensionalFunctionIntegration(UniformRandomNumberSequence sequenceGenerator,
                                  DoubleUnaryOperator integrand).
```

(a) Consider a class implementing the interface `UniformRandomNumberSequence` by a Van-der-Corput sequence with a given base. Can you pass an object of such a class to the constructor above, together with a given `DoubleUnaryOperator`, to achieve a good approximation of the integral in (3)? Give a short explanation of your answer.

(b) Is there any method you would add to the interface `UniformRandomNumberSequence` in order to get a better approximation of the integral in (3) when you pass an object of a class implementing such an interface to the constructor of `TwoDimensionalFunctionIntegration`?

Note: this is *theoretical* exercise, not a coding one. Of course if you like you can write the code in order to have a look at what happens, but that would not be the solution.