**Goals:**

- Try alternative model-based testing techniques in practice  
- Compare and contrast different model-based testing techniques

**Introduction:**

In this assignment, you will write a property-based specification of parts of the library you tested in assignment 2 and compare to that experience. (The SUT should be the API you have been using during assignment

2. Based on your property-based testing you should then answer the questions below in a project report.

For Java, we encourage you to use either the QuickTheories library [https://github.com/ncredinburgh/QuickTheories (Links to an external site.)](https://github.com/ncredinburgh/QuickTheories)

or the junit-quickcheck library [https://github.com/pholser/junit-quickcheck (Links to an external site.)](https://github.com/pholser/junit-quickcheck) but there are also other QuickCheck libraries out there which you can choose.

For Python, we recommend Hypothesis [https://hypothesis.works/ (Links to an external site.)](https://hypothesis.works/).

**Specific tasks / Questions to be answered:**

**Part 1. Design of properties**

You should write a total of at least 8 properties to test a part of your SUT. Ideally, the tested parts should be ones that were also used in the parts you tested in assignment 2 but depending on your library and what you focused on in assignment 2 you might need to deviate from this. If so explain why in detail.

Q1. Which parts did you select to write properties for and why are they more likely to be good targets for property-based testing?

Q2. Describe all of the properties you wrote. Include the purpose with each property, the code/implementation of the property as well as specific generators or setup code you needed.

**Part 2. Manual mutation testing**

You should manually mutate the code in a way that each property is triggered at least once.

Q3. Describe the mutations that you had to do and which property(ies) each mutation triggered as well as the output of the tool.

Q4. For at least 3 properties/mutations, analyse to what extent the output from the PBT tool would have helped you find the problems and debug it.

**What to submit:**

You should submit a single PDF report answering the questions above to the course submission system.

There are no hard constraints on how short or long your report can be but a good indication length is that the text itself (excluding any figures/graphs or tables) corresponds to about 3 pages in a normal 12pt font.

But don't extend your answers unnecessary just to make your result look more impressive; the key is the actual content of your answer and how relevant it is to the question.

You are encouraged to use figures, graphs, and tables to structure your responses and give details, as you see fit.

The key is that you fulfil the goals above and that your answers clearly show that you have.