Technische Hochschule Ingolstadt, WI
L. Lodes, M.Sc., Daniel Kriegl, M.Sc.

Software Development (Winter 25/26)

Assignment 05

Bonus points can be claimed until November 17th 2025, 09:25 AM

Exercise 1 *(Sets, 3 Points)*

Consider the example on slide 26 of slide set 4 "Data Structures". Here we demonstrate that searching 1000 random numbers in a list of 100 000 numbers is several orders of magnitude slower than in a set of the same size. This can be explained by the so-called runtime complexity of the underlying search algorithms. While in a list with twice the size also the search cost becomes approx. twice as large, the search cost in sets increases only in approx. log_2-fold relationship. Your task is to write the underlying code and to perform some experiments with it. To do this, proceed as follows:

- Research how to use the `random` module to draw a list and a set of a specified number of random numbers *without replacement* from a specified interval.
- Research how to measure the execution time of some lines of code using the `time` module.
- Make up lists/sets of 10 000, 100 000, 500 000 and 1 000 000 random numbers respectively drawn *without replacement* from the interval $[0, 10000000]$.
- Using the `in` command, check 1000 times whether a randomly drawn number from the interval $[0, 10000000]$ is included in the list/set. Measure the time taken in each case.

Compare the results of the different sized lists and sets. What are the effects of the two data structures?

Exercise 2 *(Guessing numbers, 2 Points)*

Write a programme that plays a guessing game with the user. In this game, the computer first chooses a random integer from an interval defined by the user. The user then has 5 attempts to guess the number chosen by the computer. If the user's guessed number does not match the randomly selected number, the computer shall tell the user whether the guess was greater or less than the number to be guessed. If the user exceeds the number of attempts, he/she loses.