0.1 (E) Structure of the exercises

This is the exercise for the lecture Protein Prediction I for Computer Scientists. Each lesson will have different sections:

- A talk to rehearse and explain things from the lecture. This may also include background material which was not covered in the lecture, but which we consider necessary and helpful.
- A discussion of the previous sheet’s programming homework
- Questions & answers
- An introduction of the new programming tasks
- Parts of the exercise might be relevant for the exam

**Infrastructure:** To offer you an individual learning experience, everybody is encouraged to submit the programming tasks. This semester, we will use the ArTEMiS system which was developed by the Chair for Software Engineering.

0.2 (E) Common Rules

- Python as mandatory programming language
- Strict compliance with the input/output specification and requested program or function names
- Submit homework to earn the bonus before Tuesday 10 a.m 2 weeks after the exercise was published, i.e for the first exercise on 14/05/19
- Submission dates can also be found on ArTEMiS
- Unless otherwise stated, all sequence positions are 0-based (starting with 0)

0.3 (E) Grading
In addition to the programming tasks for the exercise, there will be short multiple choice quizzes after every lecture about the previous lecture’s content. Each quiz will start at 11:45 and take approximately 15 minutes. Participation in the quizzes and the programming exercises will give a combined grade, which may (in case of an improvement) be used as a bonus for the final exam. The bonus grade and the grade of the final exam will each count for 50% of the final grade. The bonus can only be applied to a passing grade (i.e. exam was passed with ≤ 4.0). You need to achieve at least 75% of the overall points to be eligible for the bonus.

The grading schema of the achieved bonus points will be decided at the end of the semester and the total bonus points will consist of 80% programming exercise points and 20% quiz points.

0.4 (E) ArTEMiS
We will use the AuTomated assEssment Management System for Interactive Learning (ArTEMiS) during our exercises. Log into the system’s website at http://artemis.ase.in.tum.de/ using your TUM online credentials (use your ga99xxx TUM ID and the corresponding password). Go to the page Courses”. Click on the ‘Start exercise’ button for the corresponding exercise, then click on the ‘Clone repository’ button to checkout your personal repository containing a code template via Git. Modify code to satisfy the requirements described in the worksheet, commit your changes into the master branch and push them. Pushing your changes will trigger a test runner which checks your submission using automated test cases and you will be provided with a summary of test execution. You can submit multiple times until the submission deadline is reached. Submissions after the deadline will be disregarded by the system automatically.

0.5 (E) TUMonline
Lecture Slots:
- Tuesday, 10:30 - 12:00, MW1801, Ernst-Schmidt lecture hall
- Thursday, 10:30 - 12:00, MW1801, Ernst-Schmidt lecture hall
Please try to evenly distribute on all four exercise slots:
- Tuesday 13:00 - 14:30 – 2250, Seminarraum MW B8 (5502.02.250, engineering building)
- Tuesday 14:00 - 15:30 – MI 00.06.011 (Hörsaal 3)
- Thursday 13:00 - 14:30 – 2250, Seminarraum MW B8 (5502.02.250, engineering building)
- Thursday 12:30 - 14:00 – MI 00.06.011 (Hörsaal 3)

0.6 (H) Quick Intro to Python Basics
You will have to write Python code to do the exercises. If you do not know this language, please check out the Python tutorial at https://docs.python.org/3/tutorial/.

0.7 (H) Quick Intro to Git
You will have to use Git version control system to do the exercises. If you do not know how to use Git, please complete the quick Git tutorial at https://try.github.io/

0.8 (H) First exercise

The first exercise is mainly aimed at testing your capability to successfully work with ArTEMiS and Git. Start the exercise in ArTEMiS and clone the repository, then modify method ‘complementary’ in the 'main.py' file such that it returns a string of complementary DNA nucleobases for a given string. For example, it should return ‘T’ for ‘A’, ‘C’ for ‘G’, ‘ATGC’ for ‘TACG’. Commit your changes and, hopefully, you will see a score of 100% in the ArTEMiS web-interface.