EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

BIO-4372
CHEMINFORMATICS
Winter Semester 2022-23

Dr. Philipp Thiel
Institute for Bioinformatics
and Medical Informatics

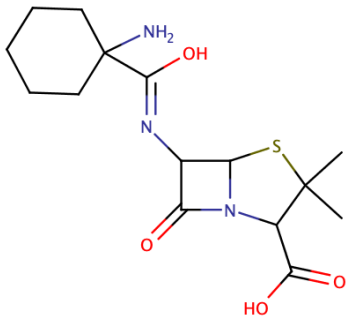
Assignment Sheet 3

Submission deadline: November 14, 2022, 9 a.m.

A3.1: Line Notations [6 points]

In this task you have the chance to practice converting from line notations to molecules and vice versa. Please either add a scan or a photo of your hand-drawn molecules to your report.

1. Derive a SMILES string for the following molecule:



2. Draw the 2D structure of the molecule given by the following SMILES:

   `O=C1CC2(CCCC2)CC(=O)N1CCCCN3CCN(CC3)c4ncccn4`

3. Draw the 2D structure of the molecule given by the following InChI:

   `InChI=1S/C7H10N2O2S/c1-3-11-7(10)9-5-4-8(2)6(9)12/h4-5H,3H2,1-2H3`

4. What are the names and indications (if any) of these compounds?
   Hint: One of the best online resources for information on drugs is DrugBank

A3.2: Implementing a SMILES Generator [14 points]

Depth-first traversal (DFT) is a recursive algorithm that is used to traverse graphs. Starting from a root node, it explores all adjacent vertices that have not been visited yet. When it is not possible to move forward in the current path the algorithm employs backtracking to explore unvisited branches. An important Cheminformatics use case of DFT is the generation of SMILES from molecular graphs.

In this task, you have the chance to implement such a method in Python using the RDKit library. For this purpose, please download the archive `chin-materials-a3.tar.gz`. It contains a working piece of Python code that reads molecules from an SDfile, iterates over the molecules, calls two functions, and writes molecule IDs and generated SMILES (default: empty string) to a TSV output file.

Additionally, the archive provides two SDfiles as input. The file `smiles_01.sdf` is useful to test your implementation because it contains only a small number of molecules and some extreme cases. The file `smiles_02.sdf` contains more molecules and the corresponding output file has to be handed in.

In order to solve this task you have to implement the functions that contain the `#TODO` flag. I can strongly recommend to use e.g. KNIME or a chemistry drawing program to check your results while developing.

The function `assign_custom_atom_id(mol)` seems probably superfluous to you. Nevertheless, I highly recommend to implement it so that it assigns custom and unique integer IDs to the atoms of the given molecule. Subsequent SMILES generation then should use these IDs to guide the DFT. You probably already have an idea why this could be important ... and you are right: for assignment sheet 4 :)

Important notes: There are no molecules with rings and you do not have to implement ring processing, charge processing, as well as processing of disconnected structures! Do not forget to comment your code. It should be understandable, clean and reproducible.

Bonus Points: If you feel bored and have time left you can try to enhance your solutions by correct processing of charges (+2) or by correct processing of disconnected structures (+2).

Submission

1. Your PDF report with the solutions for A3.1.
2. Your properly documented Python implementation.
3. Your output file generated for `smiles_02.sdf`

Please bundle the files in a tar-archive to simplify uploading to ILIAS.

▷ Please use Slack to discuss problems in the first place
▷ If you have confidential questions, don't hesitate to drop by or write an e-mail