ChemBE 414/614 Homework #2 Due: 2/14/2018

For these and all coding problems assigned in class, submit your code and code output in your pdf single-file upload to Blackboard.

Your code should be easy to read; have explanatory names for variables, objects and functions; have docstrings (help messages) for functions and objects; have comments; and follow the Python coding standards (PEP-8) as much as possible. Begin writing your code using the provided templates on the course website.

1. [30 pts] *Torsion space calculations*

   a. [15 pts] Calculate the coordinates of the N atom of residue *i*+1, given that residue *i* has $\varphi = -126.1^\circ$ and $\psi = 154.4^\circ$. Use bond lengths and angles from Engh & Huber *Acta Cryst.* (1991). To make the calculation simple, place the carbonyl C atom at the origin, the C_α on the negative *x*-axis, and the first N in the *xy* plane with a positive *y* coordinate. For completeness, also give the coordinates of the C_α and N atoms.
   b. [7 pts] How would you continue placing atoms along the backbone (*i.e.* without the fortuitous placement of the C at the origin)?
   c. [10 pts] **Bonus**: also calculate coordinates for residue *i* for the C_β, C_γ (use $\chi_1 = 62.0^\circ$), and carbonyl O atoms.
   d. [8 pts] To check your answer, create a PDB file by hand with the four (or five) atoms and their coordinates. The PDB file format can be found at http://www.wwpdb.org/documentation/file-format. Use the "ATOM" record, and you must complete at least the following fields: *name*, *resName*, *resSeq*, *x*, *y*, *z*. Be sure to place the right information in the right columns, since the data are input strictly by column-number. Once you have typed the PDB file, load it into PyMol to confirm that your atoms look correct. You can use the measurement wizards to check your angles and distances. Submit the pdb file listing and a picture for credit.

2. **[30 pts] Torsions to Cartesian Python Program.** Without using PyRosetta, write a computer program which can calculate the backbone atom coordinates of an arbitrary-length peptide when given the peptide length and the set of φ and ψ angles. The input should be loaded from a file called 'torsions.dat' that has the number of residues on the first line, each pair of ϕ and ψ angles on subsequent lines. Your program should assume standard bond lengths and angles (from Engh & Huber) and $\omega = 180^\circ$, although you might write the program such that you can relax these assumptions if needed. The Python package NumPy may be helpful for vector operations.

   The following reference may be helpful: Parsons *et al*. 2005 *J. Comput. Chem.* 26:1063-1068. Note the two typos in the Parsons article: First, the *yz* term in the matrix on the bottom right of p. 1065 should have a $(1 - c\theta)$ term rather than a $(1 - s\theta)$ term; second, on the top right of p. 1066, the *x* term of $\vec{D}_2$ should be $-R\cos\theta$.

   Use PyMol and/or PyRosetta to verify your calculations.

3. **[20 pts] Workshop #2 Programming Exercise 2.** Submit a program called ‘make_helix.py’ and the PDB file. The program make_helix.py should input an argument to specify the number of residues in the helix. The program should check that the user has entered a reasonable value, and if not, give a meaningful error message. Your program must have a docstring and comply with the PEP8 style guidelines as much as possible. The goal is to make the code easily readable and understandable. Submit the short answer response with the rest of your homework.

4. **[25 pts] Workshop #2 Programming Exercise 4**. Again, your program should have a docstring, check input values, and follow the PEP8 style guidelines. To test your program, use PDB 1M40 (TEM-1 β-lactamase) to calculate the alanine propensities. Five bonus points for each bonus level (‘ss_propensities.py,’ where the argument can be a PDB file or a text file with a list of PDB files). Submit code and output.