Nitesh Turaga ,nturaga

viewMol + Amino acid explorer

Initially I wanted to make some bioinformatics application in my project, but later decided that I want to make a tutorial for bio-chemistry students to learn about amino acids easily and also learn about importing files from the protein data bank which they can view in 3D.

I used the module vPython (visual python), which helped me draw 3D structures. It was a very hard module only because there is very less flexibility in this. I started appreciating Tkinter after using this module.

vPython relies heavily on vector math especially for getting the axis of the 3D structure correct. This took some getting used to. I was first building the amino acids for which I wanted to have with a good GUI. Building the amino acids took a lot of time as they needed to be coded in with the correct axis and positions. The idea of being able to import Pdb files and making molecules was also implemented.

PDB stands for protein data bank. The user, if he downloads a protein of interest from [www.pdb.org](http://www.pdb.org) onto the Desktop, can view it in 3D as a complete molecule. This is how the popular molecular visualization tools have been designed like RASMOL and PYMOL with some added functionalities.

The molecule viewer was hard to come up with. It needed a good knowledge about how to handle buttons and also how to get inputs from the user about files. The design was simplified from what I had imagined earlier. I wanted to have a user interface with a complete file browser, but I changed this to a much simpler GUI after receiving multiple inputs from the people. Now the design stands in a way that I made my own buttons and used Tkinter for its “simple dialog” and “message box”.

This was pretty hard for me as the code was complicated and we did not handle file I/O in class. Once the filename input was received from the user the real issue came up. Reading a PDB file was hard, in the sense that it has the co-ordinates of the protein, the ligand, the hetero-atoms etc. This viewer is now designed to view only the protein part of the molecule which is the most important part.

I placed the Tkinter box separately on the left side of the screen, so that the viewer can view the Tkinter GUI on one side and the vPython canvas opens on the other side.

Viewing the structure of the protein is helpful in research in chemistry and biology in various fields. The program is designed to take only the co-ordinates of the Atoms by going through the entire file and storing them in a list and making each atom with those coordinates. The challenging part was connecting those atoms. The distance between one atom to the other was something the PDB file information did not contain. This did not support the vector position and axis format used by Vpython.

Another problem encountered during my program design was that, the vPython module works in a way it can only “make “ the atoms. It cannot delete them like in Tkinter where we can have the delete(ALL). This proved to be a huge problem further on in my program. I struggled to get the basic functionality of my Tutorial also working because of this. The mouse clicks could not be programmed properly because of this feature in the vPython module. I ended up initializing too many global variables to control the visibility phenomenon of the amino acids.

I used a lot of helper functions to make the drawing of Amino acids easier using vPython which proved to be a “bane”. Fixing the visibility issue took me two entire days and this was one of the main reasons I could not implement better functionality in my code for the molecule viewer by adding cooler functions. Some of the functions I planned to have are : if I click on the atom displayed in the Vpython canvas, the name of the atom would come out in the Tkinter box or the console. Such features are really useful in molecular visualizers.

The GUI was built on my own completely. I have drop down options which were challenging and some fancy bugs popping up every now and then unexpectedly. But the drop down options are fully functional and work with the on – off technique. The user presses once to view the amino acid and then clicks again to make it disappear. This is the simplest way I could design the drop down boxes. Although it might not have been extremely user friendly.

The coolest feature for sure is the molecule viewer and is helpful for teaching high school kids and undergrad students in chemistry classes about amino acids but simply loading them from pdb if necessary.

The best part at comes at the end where I managed to optimize my code by thinking about the algorithm I was running to make the protein view from pdb. I cut down some loops and now the program is lightning quick to load in pdb files of any size.

GUI issues and Fixes:

The GUI is pretty cool in the sense it was made completely from scratch using Tkinter only. The simple dialog and the message box feature were used for very small details of the GUI and nothing heavy. The entire GUI woks on the usage of on-off functionality. For all the areas where there can be mouse click events there needs to be one click to turn it on and one click to turn it off.