Exercise Ex5

Student: Firstname Lastname Sciper: 000000

**Please use this template to submit your answers.**  
If you had to modify code from the notebook, please include the modified code in your submission either as screenshot or in a

\begin{lstlisting}[language=Python]  
\end{lstlisting}

environment.

You only need to include the code cells that you modified.

Note, that references to other parts of the documents aren’t resolved in this template and will show as ??. Check the text of the exercises on website for the reference

**Exercise 1**  
Implement the initialization described in the Theory section in our ToyMD code in toy\_md.py in the section ##INITIALIZATION HERE##. For this you should pick random velocities using r.random.standard\_normal((dim1,dim2)) which will produce random numbers between 0 and 1. You need to shift this random gaussian distribution appropriately and then multiply it such that the width of the velocity distribution matches the kinetic energy at the target temperature. Use the variables masses[i] for the mass of particle i and Boltzmanns constant 0.00831415. Remember that for each degree of freedom (e.g velocity in x direction) ([[equipart\_x]](#equipart_x)) holds.

Your answer here

**Exercise 2**  
Implement the Berendsen thermostat in the toy\_md.py and toy\_md\_integrate.py (change the compute\_lambda\_T function) files.

Your answer here

**Exercise 3**  
A better thermostat is the Andersen thermostat. It can be implemented as follows. Describe what problems this thermostat will present to us e.g for sampling of diffusion coefficients. What advantage does this thermostat have compared to the Berendsen thermostat?

#Andersen Thermostat  
for i in range(N):  
 if (r.random()<float(md\_params["tau -T"])\*float(md\_params['time -step'])):# pick random particles according to nu  
 sd=np.sqrt(0.00831415\*float(md\_params['temperature'])/masses[i]) # Velocity distribution at target temperature  
 velocities[i]=0+np.random.randn(1,3)\*sd # set new random velocities

Your answer here

**Exercise 4**  
Visualize the trajectory and visualize the distance of a C-O bond using the code cells below. Explain the fluctuations that you observe. What does the average value correspond to?

Your answer here

**Exercise 5**  
Visualize the radial distribution function of this system using below code. What do you observe?

Your answer here

**Exercise 7**  
Test the influence of the coupling parameter and timestep . Test the combinations of:

* large and small
* large and small
* very large and

For some of these settings the system may explode (you can see that because really high energies and temperatures are reached and the simulation gets stuck). If this is the case, you will not be able to complete the simulation, but you can just restart the notebook and use new parameters. **In your report, even for those parameters add a comment and explain the behaviour you observe**.

**Hint**: To check the influence of the coupling parameter , you can plot the instantaneous temperatures for several runs using different values. For this, you need to rename the logfile produced after running toy\_md (to avoid overwritting it with the next run). You can do this by executing this command on the terminal: cp <path\_to\_logfile>/logfile <path\_to\_logfile>/logfile\_tau\_<value\_of\_tau>. To then read the temperatures from each logfile, you can use the function read\_temperatures shown below.

Note that the code cell for plotting already suggests you a possible set of values of and , adapt the values of the parameters and plot different/more curves.

Your answer here