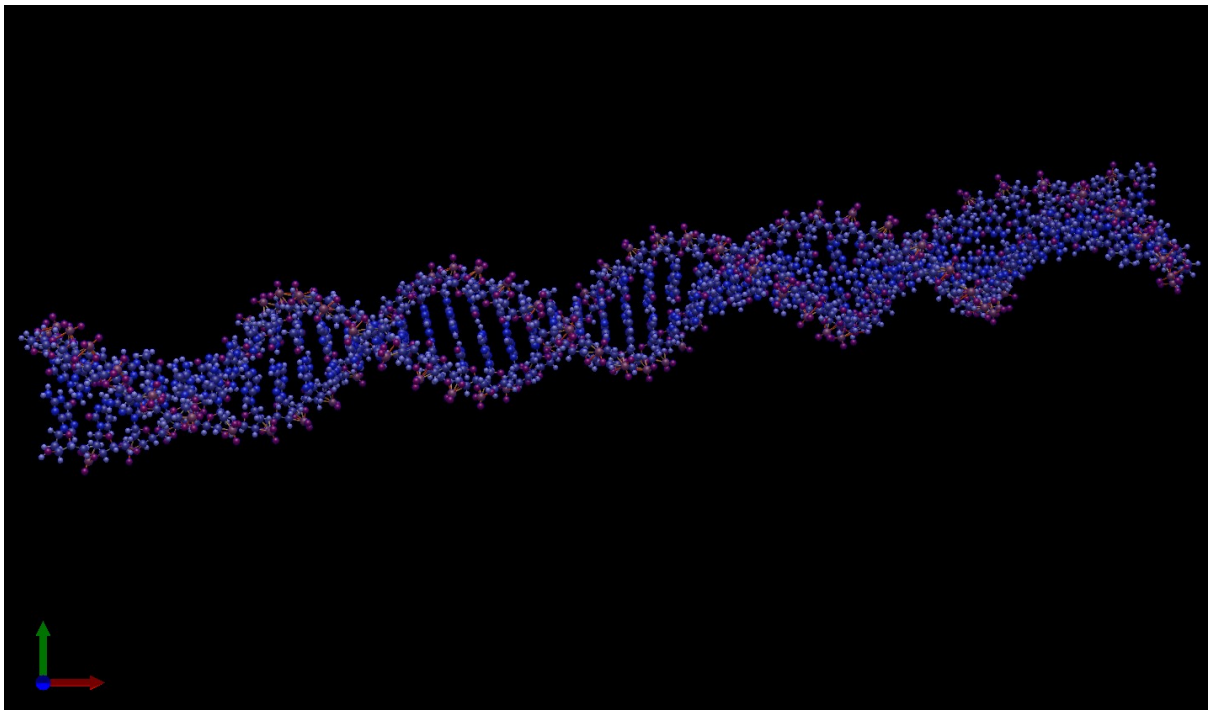
***Disclaimer***

*I strongly invite you not to use LLMS (chatGPT or so) for this task. Please do not make me read code generated, or even just polished by a bot, it would be a missed opportunity for you to learn how to approach a problem and how to code it (that means, to be really able to solve it). I look forward to seeing how much you learn in the class, what has been clear and what has not been clear.*

## ***Assignment 3 (****Deadline **9***. November 23:59)

***TOPIC 1: Calculate DNA volume via Monte Carlo simulation.***



*Task 0: Define your simulation box (x, y, z) dimensions in a 3D space. Choose the right units! [1]*

*Task 1: Make a function to randomly position a point in your simulation box from a uniform distribution. [4]*

*Task 2: Position a sphere in a random place and size in your simulation box. [5]*

*Task 3: Make a function that checks if a point is inside the sphere or not. [5]*

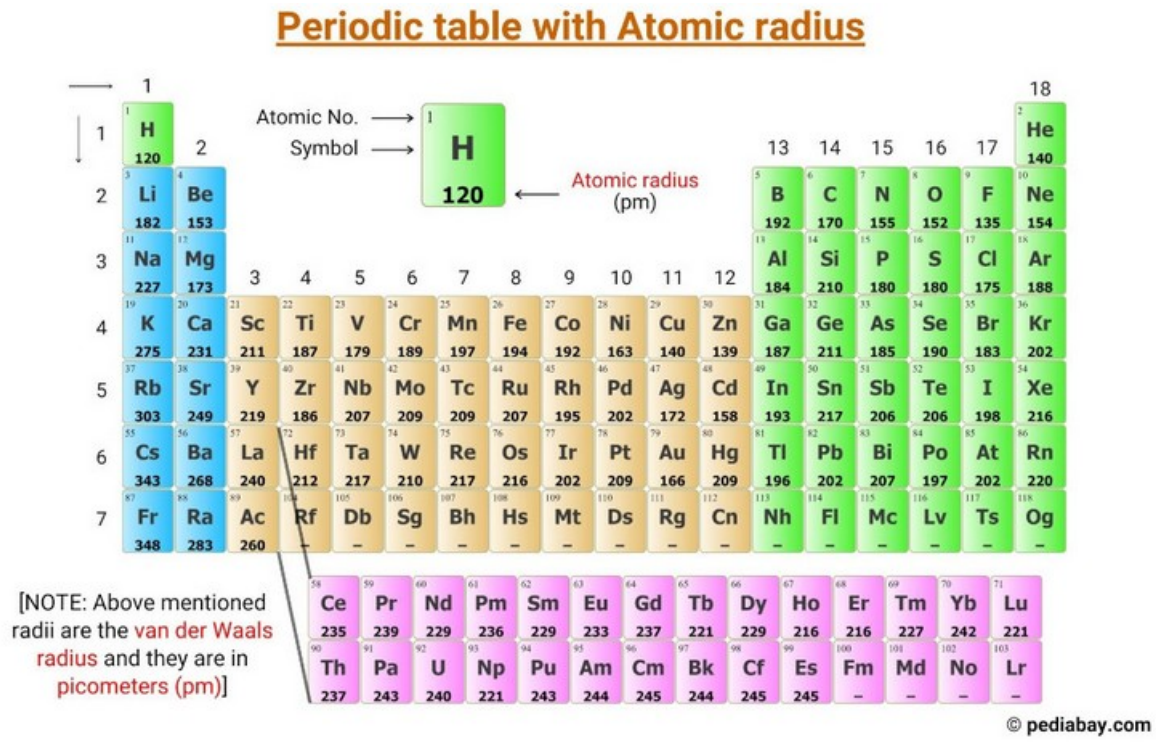
*Task 4: Calculate and plot the fraction of points INSIDE the sphere divided by the number of randomly generated points. How do you check if your result is right? [5]*

*Task 5: Calculate pi as a function of the number of randomly generated points. [5]*

*Task 6: Generate 10 spheres of random size and position them randomly in your box. [5]*

*Task 7: Calculate and plot the fraction of points INSIDE the sphere divided by the number of randomly generated points. How do you check if your result is right? [5]*

*Task 8: Read the attached file containing the coordinates (in **angstrom**) and the atom type of the DNA molecule. Use the following periodic table of elements to find the dimension of each atom (each sphere). [10]*




*Task 9: For each atom now you have x, y, z position and its dimension (make sure that the units are consistent!). Make sure that the simulation box contains all your system and that it is not too big. [5]*

*Task 10: Calculate and plot the fraction of points INSIDE the sphere divided by the number of randomly generated points. You just have calculated the Volume of DNA!! How do you check if your result is right? [10]*

***TOPIC 2: Random walk for accessible volume calculation***

*Task 1: Make a function to generate a set of random walkers in 3d starting from a set of different random points [2]*

Task 2: Make a fast function to generate a set of random walkers in 3d starting from a set of different random points [3]

*Task 3: Describe a strategy to calculate the accessible volume of DNA. Note that this can be done in different ways. [10]*

*Task 4: Describe a test to verify if your strategy is correct. Hint: a correct answer might not exist, you might thus design an approach that is checking if your outcome is within a meaningful range [10]*

*Task 5: Code and test your approach. [20]*

***Deliverable:***
*Please put all the functions (and classes) you have generated in a .py file.*

*In a jupyter notebook, instead, import and use your functions (and classes) to calculate the desired results.*

***Delivery method:***

*Git: make your own repo. If you prefer to keep it private, send me an invitation link to me (EnricoRiccardi) and to the TAs (ask on Discord or in presence for their account name). If you want to keep it public, just share your repo link as a reply to this post.*

*In the jupyter notebook, use* ***TOPIC 1:*** *task 1, task 2, etc to give the answer to each point using Markdown syntax. Please use, for each task, one cell to help us going over your work.*

***How will this project be graded:***

*105 points available. Each question has a number of points available indicated in [].*

*Your final score = total points accumulated + quality_factor * bonus points*

*quality_factor = total points accumulated/max point available*

*If AI is used the quality factor will be equal to zero. If AI is used and not stated, it is considered cheating and the project will be given 0 points.*

A grade for the assignment between A and F will be given.

***Bonus points:***

*Use* ***pylint*** *on your function file. Pylint gives a score from 0 to 10. That will be your bonus points that will be applied.*

*Up to 5 extra points for proper use of* ***assert*** *statements in your code.*

*1 extra point for each function that tests if your code works correctly (max 5 points).*

*The shorter the jupyter notebook in terms of number of lines, the more bonus points you will get (Up to 5 points). Comments and assert statements do not count in the line numbers.*

*For each day earlier that you deliver, you will get two extra points. For each day of delay, -10.*

***Group work:***

*Groups of 3 are suggested.*

***Other Requirements***

*A statement on the eventual usage of AI has to be included.*

*A statement reporting the Individual contributions has to be written.*