BF550: Programming in Python, Fall 2020
Problem Set 0 is due by 5 pm on Monday, September 14

Submission
To help us managing the homework, please (i) submit your assignments by email and add the following subject line: "BF550: HW0" and (ii) start your file names with your last name, for example, "korolev_hw01.py". I ask that each homework is submitted as a single ipython notebook that shows the code and some test examples demonstrating that the code is working. In addition to the notebook itself, please submit the notebook converted to PDF. You can convert ipython notebooks to PDF from their menu. Sometimes this doesn't work. If so, don't worry and just submit the notebook itself. Please send your emails to Howard and cc me.

Reading Assignment
Using Python documentation (`https://docs.python.org/3.8/`) or any other source familiarize yourself with standard arithmetic operations, boolean operations, functions `print()` and `input()`, and the methods of the string class (`class str`). In particular, familiarize yourself with the method `format()`; the following website could be helpful `https://pyformat.info`. You will also need to review the `for` loop, `if` statement, lists, and dictionaries.

Problem 1
Implement a function that takes a string of DNA base pairs and outputs a complementary sequence (in the same order). For example, `'GATTACA'` should be mapped to `'CTAATGT'`. Then update the function so that it can take both upper and lower case letters in the input string. Then, update the function with a default argument `direction='same'`. When the user explicitly sets the direction to `'reverse'` the function should reverse the order of the base pairs. For example, `'GATTACA'` should be mapped to `'TGTAATC'`.

Problem 2
Write a program that asks user to enter an integer and then reports whether it is a prime number or not. The script should run from `Python` shell when called by `run prime.py`.

Problem 3
Write a module that contains a function `sqrt(y, tol=1e-6)`, which computes a square root of a number using Heron's algorithm with guaranteed relative error less then `tol`. The module should run as a program that asks for user input and prints output when executed using `run sqrt.py`.

Heron's algorithms for finding x such that $y = x^2$ works as follows. First, you come up with an initial guess for x; think what it should be. Then, you update x using the following formula:

$$x_{\text{new}} = \frac{1}{2}\left(x_{\text{old}} + \frac{y}{x_{\text{old}}}\right).$$

Computation ends once the relative deviation between x^2 and y is less then the required value.