Academic honesty and plagiarism

- Plagiarism is unacceptable.
  - Code you submit must be your own. No copying, adapting, or submitting code you did not create is allowed. Working together and presenting variants of the same file is not acceptable. Here are some specific guidelines to make sure you don't cross the line:
    - Do not exchange programs or program fragments in any form on paper, via e-mail, photos, or by other means.
    - Do not copy solutions from any source, including the web or previous quarters' students.
    - Do not discuss code with other students at the level of detail that will lead to identical programs or program fragments.
    - Do not use chatGTP or other AI tools
  - Ask me if you are uncertain.
  - **All violations of the academic honesty will be immediately referred to the dean's office.**
- **Plagiarism detection will be applied to all code. It is very good and will catch you!! (20% of a class referred to dean!!)**

Lab 0 Sort

Due April 4, 7:00pm CST

You are tasked with updating the function process(data,stats) in student_code.py. This code takes the array "data" passed into the function, and re-orders the data in ascending order (smallest numbers at lowest index, highest at highest index). For example, it changes this [-1, 3, 2, 8, 9, 11, 100, 4] to this: [-1, 2, 3, 4, 8, 9, 11, 100]. It also update the array stats with the [mean, median, mode] based on data from the data array. When you submit your code on canvas, you should only submit your student_code.py. (do not worry if canvas changes the file name after submission)

You cannot include any additional modules!!

The goal of this lab is two fold:

1: It is designed to familiarize everybody with environment and submission process for the class.

2: It is designed to make sure you are comfortable with using python -- If you struggle with Lab0, you don't have the python coding experience needed for 348, and will struggle with the remainder of the labs, which only get more difficult. If this is the case, I would recommend dropping the class.

Some considerations for this lab:

-Data array can have any size >= 1

-Data array will only have integers, ranging from -1000 to +1000 (inclusive)

-You cannot use the python built-in list function sort() or sorted(). Code that uses this will receive a 0.

-You cannot include any additional modules

-Mean should be rounded down to the closest integer.

-In case of even array size, median should be the lowest of the middle values.

-In case of ties for mode, return the value with the most occurrences which is the lowest.

-Make sure you use python3, and that your code runs correctly on the Northwestern servers described in class.

Execute and test by running: python3 main.py

- This tests your code against some test cases, but passing all test cases does not guarantee you will pass all grading cases (they will be different). Try your own test cases too! Think about corner cases.