EEE 419/591
Spring 2026
Homework 1

Solve the following three problems. Put them all into a single code named hw1.py and submit it to the designated link on Canvas. If you upload a second attempt to Canvas and it renames your file, ignore the renaming.

In this HW, only limited AI is allowed. Examples of allowed prompts on this HW:

- What is a greatest common divisor of two numbers?
- How do I write an if statement in python? Give me an example
- How do I check if a number is integer in python?
- I got this error message. How do I interpret it?

Examples of what unallowed prompts on this HW:

- Write me a code that calculates the greatest common divisor of a list of numbers in python
- I ran this code. [PASTE CODE HERE]. I got this error message. [PASTE THE ERROR MESSAGE HERE]. How do I fix my code?

Do not forget to cite your AI chat if you used any. Check the last page of this HW assignment.

Problem 1: Greatest Common Divisor

Write a code that takes a list of integer numbers and gives the greatest common divisor of them all. Do not use the "greatest common divisor" function built-in in any of the python packages.

Your code should expect a list of numbers as an input and print "The GCD is: " followed by the greatest common divisor of the list passed to it by the user. If one of the numbers is not integer, then your code gives an error message, ignores that number and finds the gcd of the remaining numbers. Keep in mind that a number formatted as float does not mean it is not integer.

An example of output of your code is as follows.

```
Input a list of integers: [24.4, 60, 120, 30.0]
Error: Ignoring 24.4, number not integer
The GCD of the integers in this list is: 30
```

Hint: start writing a code that works on a list of 2 numbers.

Problem 2: Prime Number Checker

Write a code that takes a list of integers and outputs, for each number, whether it is prime. Do not use the "prime number checker" function built-in in any of the python packages.

Your code should expect a single list of numbers (possibly not integers) as an input and prints if each number is prime. If one of the numbers is not integer, then your code should skip that number and let the user know it is not integer. Keep in mind that a number formatted as float does not mean it is not integer.

Consider the following example:

```
Input a list of integers: [31.0, 17, 81.4, 28]
31 is a prime
17 is a prime
81.4 is NOT an integer. Skipping ...
28 is not a prime
28 is not a prime
```

Problem 3: Reimann-Zeta Function

Write a code that calculates an approximation to the Reimann-Zeta function for a specific value of integer. Your code takes a single integer and outputs an approximate value of the corresponding Reimann-Zeta function of that integer as well as the number of terms your code depends on to calculate that function. Only print up to the first 4 decimal places of the function's value at the given input. Do not use the "Reimann-Zeta" function built-in in any of the python packages.

Recall that the Reimann-Zeta function is given by

$$\zeta(s) = \sum_{n=1}^{\infty} \frac{1}{n^s}$$

Since we cannot calculate an infinite sum of terms, you will need to use only a finite summation which is why your code will give an approximate value, no matter how many terms your code computes. It is up to you to decide how many terms your code computes to give good approximation. It is up to you to define how good is "good". Your code should print something as the following example.

```
Input an integer: 2
Zeta(2) = 1.6449 based on 10,000 terms
```

MANDATORY – Citing AI chat in each assignment, if used:

- Using AI: If you use AI in an assignment, you MUST cite the chat that you used to help you finish that assignment. Keep in mind the following:
    - Cite your chat by pasting its link in your code as a commented line. If you use multiple chats on the same or different AI platforms, cite them all.
    - Make sure you use an AI platform that allows sharing a link to the chat and that allows the chat to be viewed publicly by anyone who has the link.
    - Each assignment needs to have a separate dedicated chat. Parts of a single assignment can share the same AI chat or each can have its own chat.
    - Each chat needs to have the following identifying info: Your name, Assignment name, today's date. The following prompt to the chat is sufficient for the identifying info to be saved in the chat:
      Name: First Last Name
      Assignment: HW 1 – Prime Numbers
      Date: Sunday Jan 24, 20xx.
    - A chat without identifying information means it is not yours. The identifying information could go at the beginning or at the end of the chat.
- Not using AI: If you do not use AI at all in a particular assignment, you must write the following commented line in the code you submit "*I did not use AI at all to complete this (part of the) assignment*".