# PRACTICE WITH LOOPS

CS 125

**Working with a partner/group, use the following steps to solve each of the following problems.**

(a) Plan your function on the white board (either on the classroom wall or on Zoom). Write out your entire program. Think about what errors might occur and how to fix them.

(b) Plan multiple test cases for your function. What input will you send to your function? What value should the function return?

(c) *Only after you have completed steps (a) and (b) should you type your code in Python.*

(d) After you have typed your function, run your test cases. Does your function work? If not, how can you fix it?

1. Write a function `computeSum(n)` that accepts a number $n$ as a parameter and uses a loop to figure out how many terms in the sum $1 + 2 + 3 + \cdots$ are necessary for the sum to exceed $n$. Your function should return the number of terms. For example:

   `computeSum(6)` returns `4`

   `computeSum(17)` returns `6`

2. Write a program that prompts the user for an integer between 1 and 10. However, if the user enters something that is *not* an integer between 1 and 10, your program should print an error message and prompt the user to try again. Your program should repeat this as many times as necessary, until the user enters an integer between 1 and 10.

3. Write a program that asks the user for a string of lowercase letters and uses the `isVowel()` function from the previous class to determine whether the string consists entirely of vowels.

4. Use a loop to investigate the following sum:

   $$1 + \frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \frac{1}{16} + \frac{1}{32} + \cdots$$

   What happens when you add up many terms of this sum? How many terms are necessary to obtain a sum greater than 1.99? How many terms are necessary to exceed 1.9999? Do you think the sum will ever exceed 2?