CS2065 Spring 2022
Discussion: Iteration and Conditionals (Lab 05)

Welcome to Lab 5! This week we will be discussing conditional statements and iteration, which are powerful computational tools that we will use throughout the course. Conditional statements allow data scientists to make more complex decisions with their code, while for loops allow us to repeat the same action many times.

Question 1. What does the following function do? Fill out the docstring description for the function (the first line). *Hint: try to figure out what the function would do on different inputs.*

~~~
def mystery_function(n1, n2):
      ```___________________________________'''
      if n2 - n1 > 0:
            return n2 - n1
      elif n2 - n1 < 0:
            return n1 - n2
      else:
            return 0
~~~

Question 2. The instructor of a lower division statistics class has assigned you a task: make a function that takes in a student's score on a scale from 0 to 100 and assigns a letter grade based on the following grade boundaries.

Score	Letter Grade
0 - 69	F
70- 79	C
80 - 89	B
90 - 100+	A

Complete the function `compute_letter_grades`. It takes in a student's score and returns the letter grade they should receive.

```
def compute_letter_grades(score):
     ‘’’
     compute_letter_grades(10)
     >>> “F”
     compute_letter_grades(99)
     >>> “A”
     ‘’’
     if ______________________________:
          return ______________________________
     elif ____________________________:
          return ______________________________
     elif ____________________________:
          return ______________________________
     else:
          return ____________________________
```

Question 3. Skeleton code for the function `count_evens` is below. The function takes in an array of numbers and returns the number of even numbers in the array.

a. Use a combination of iteration and conditionals to complete the function below.

Hint: the % operator returns the remainder if you divide by a certain number! Example: 11 % 5 = 1

```
def count_evens(n_array):
    num_evens = _____________________________

    for _____________________________________:
       if _____________________________________:
           ______________________________________

    return ______________________________________
```

b. Use array operations to complete the function below.

```
def count_evens(n_array):
    remainder_array = _____________________________________
    return _______________________________________
```

Question 4. Complete the function `separate_numbers`, which takes in an array of numbers and a boolean value. It should return the number of even values in the array if the argument `return_even` is True, or the number of odd values in the array if `return_even` is False.

Hint: Use the `count_evens` function you defined above!

```
def separate_numbers(n_array, return_even):
    num_evens = ____________________________________
    if __________________________________:
        return _____________________________________
    else:
        return _____________________________________
```