General Instructions:

- **Lab is due online by 11:59 pm of the due date.** The assignment must be typed, not handwritten or scanned.

- Label your Python programs `q<num>.py`, where `num` is the question number. For example, your solution to the first question, will be stored in the file `q1.py`.

- Make sure you understand everything in this lab before getting started. Also, make sure your programs match the output exactly as given for each question. This is important as one of the keys to being a good programmer is attention to details.

- If you forgot to bring a computer, you can check out a temporary laptop from ZACH 383 for a few hours (depending on the availability).

- Before you leave the lab room make sure to checkout with the TA by showing your student ID.

- Grading will be based on correctness and clarity. **Copying work from another source and submitting it as your own is plagiarism.** The minimum penalty for plagiarism is a zero on this lab.
Lab Questions

1. Reverse and add numbers. Write a Python program stored in a file q1.py that implements the following procedure to produce palindromic numbers.

   a) Take any positive integer \( x \). If \( x \) is a palindrome, stop. Otherwise, continue to step (b).
   
   b) Reverse the digits of \( x \).
   
   c) Add the reverse to \( x \).
   
   d) If the sum is a palindrome, then stop. Otherwise, let \( x \) represent the sum, and repeat steps (b) through (d).

Steps (b) through (d) represent a single iteration. Below, are the number of iterations required for a few numbers to become palindromic using the reverse and add procedure.

   - 8 is a palindrome. It is palindromic after 0 iterations.
   - 56 becomes palindromic after 1 iteration: \( 56 + 65 = 121 \).
   - 57 becomes palindromic after 2 iterations: \( 57 + 75 = 132, 132 + 231 = 363 \).
   - 59 becomes palindromic after 3 iterations: \( 59 + 95 = 154, 154 + 451 = 605, 605 + 506 = 1111 \).
   - 89 takes an unusually large 24 iterations (the most of any number under 10,000 that is known to resolve into a palindrome) to reach the palindrome 8813200023188.
   - For 196, it is not known if the above procedure leads to a palindromic number.

Your program will consist of two user-defined functions: `reverse_add(low, high)` and `main()`.

The `reverse_add()` function implements the reverse and add numbers procedure described above for all integers in the range low to high. For example, if low is 10 and high is 50, then the function would run the reverse and add procedure on the numbers 10, 11, ..., 49, and 50. Or, the user could be interested in a single number such as 89. In this case, low and high are both 89.

For each number in the range of interest, your function will either output the number of steps required for it be palindromic or report that it does not lead to a palindrome within 100 steps. If the low and high range includes a single number (i.e., 190 to 190), then your program will show each step of the reverse and add procedure. See examples for more details.

The `main()` function drives the program. q1.py has been provided, where the code for the `main()` function is provided for you. You will provide the code for the `reverse_add()` function.

Example. The user input (which is handled in the provided `main()` function) is shown on lines with the `>` symbol. The code you write for the `reverse_add()` function will provide the output for the lines that are not prefixed with the `>` symbol. On line 1,
the user wants to know if the integers between 10 and 15 lead to a palindrome (line 1). All of the numbers in this range lead to a palindrome in 1 step or less (lines 2–7). Next, the user wants to know if the integers in the range 84 to 91 lead to palindromes (line 8). The resulting output is shown in lines 9 to 16.

Line 17 shows that the user is only interested in a single number, 190. In other words, the range is all numbers between 190 and 190. When the range between the low and high integers is 1, then the output shows each step of the reverse and add procedure (lines 18–24) and shows a summary of the result (lines 26). For integers in the range 195 to 200 (line 27), every number except for 196 leads to a palindrome (lines 28–33). The user enters 76 (line 34) and the entire calculation is shown (lines 35–38). Finally, the user then quits the program at line 39.
2. $143 + 341 = 484$

76: PAL (2 steps)

> quit
2. Simulation.
   Note: this is not a programming problem.
Answer the following questions and write your answers in a file q2.txt. You must show all calculations to get credit.

2.1. Suppose you roll a fair die once.
   • What is the probability of getting an odd number?
   • What is the probability of getting at least 3 or more?

2.2. Suppose you roll a fair dice and flip a fair coin together.
   • What is the probability of getting a 6 and a head?
   • What is the probability of getting a tail and an even number?

2.3. A lottery bowl contains cards labeled 1 to 20.
If you randomly pick one card from the bowl of 20 cards:
   • What is the probability that you pick a card number that is multiple of 3?
   • What is the probability that you pick a card number that is multiple of 5?
   • What is the probability that you pick a card number that is multiple of 3 or 5?
Submitting Your Assignment

Once you have completed your programs, submit each of them (q1.py, q2.txt) electronically. You may resubmit your files as many times as you need until the due date. Only the most recent submission is graded.
You are required to include the following lines in the header of all your files:
# File: filename.py
# Author: Student name
# Date: xx/xx/2019
# Section: Student section number
# E-mail: student_email@tamu.edu
# Description:
# e.g. This program asks for ...

Submit your files on: gradescope.com