Repeating Code Review

Write programs that accomplish each task. Use appropriate variable names and prompts for user input when necessary. Include a header (comments) for each program, consisting of your name, student number, date, and brief description of the program, like the example below.

```
# J. Garvin (123456)
# 2020-09-30
# Given two positive integers, determine their sum.
```

Ensure that all values input by the user are within acceptable ranges, using some form of input validation. You may assume that all values will be the correct *type*.

1. Read a string of arbitrary length from the user and determine the number of E's it contains (either upper- or lowercase).

2. Write a program that simulates rolling two dice 1000 times, and counts the number of times a sum of 5 is rolled (e.g. 1 and 4). Using your data, estimate the probability of rolling a sum of 5 using the formula below.

$$\text{probability} = \frac{\text{number of fives}}{\text{number of rolls}}$$

   How does your value compare to the theoretical probability of 1/9?

3. Have the user enter a value, *n*, between 1 and 20, then display the factors of all values from 1-*n*.

4. Generate random capital letters until a "P" is produced. Then until a "Y". Contunue until the entire word "PYTHON" has been generated. Count the total number of letters that were generated. *Extension*: if you solved this using six separate loops, try to do it again using only one.

5. Implement the dice game Pig, or any of the variants listed on the page. You will need to use nested loops here to control both the game and the individual rounds within it.