Intro Practice Problems

These are similar in style to competitive programming problems, but are simpler and don't use any trickiness or hard algorithms.

A warmup to get practice with problem solving in Python.

Quadratic Formula

A tiny exercise to review Python syntax.

Write a function that implements the quadratic formula in Python. For simplicity, just do plus, not plus or minus.

$$\frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Questions to ask yourself:

- What parameters should the function take?

FizzBuzz

Output the numbers from 1 to 100, one per line, except:

- If the number is divisble by 3, instead output "Fizz"
- If the number is divisible by 5, instead output "Buzz"
- If the number is divisible by 3 and divisible by 5, instead output "FizzBuzz"

Odd numbers

Functions:

- `is_odd(x: int) -> bool`
- `print_odds(n: int)`
  - Print out all the odd positive integers less than or equal to `n`.

First write the code for the functions, then write the code for reading input and calling the `print_odds` function.

Examples:

Input:

```
5
```

Output:

```
1
3
5
```

Input:

```
8
```

Output:

```
1
3
5
7
```

Integer division algorithm

Freebie.

TODO: Introduce pseudocode.

Factoring

Now try to write some pseudocode yourself.

(harder) Print out a multiplication table

Print out a 12 x 12 multiplication table using for loops.

Output:

```
1 2 3 4 5 6 7 8 9 10 11 12
2 4 6 8 10 12 14 16 18 20 22 24
...
12 24 36 48 60 72 84 96 108 120 132 144
```

Extension: Add a function parameter `n` and print out $n \times n$ table, so can support multiple different table sizes with one function.

List/string problems

Strings are very common in competitive programming, so we need to learn to deal with them. Also lists are common throughout programming, as we deal with large (maybe we don't know how large ahead of time!) amounts of data.

Find minimum of a list

Write a function that takes a list of numbers, and then returns the smallest number in that list.

Find maximum of a list

Write a function that takes a list of numbers, and then returns the largest number in that list.

Find minimum and maximum of a list in one pass

Find both the minimum and maximum of the list, but look for them at the same time, only looping over the list once.

Then, return a list of both the minimum and the maximum:

```
return [minimum, maximum]
```

Count the number of e's in a string

Input is one string per line.

For each string, print out the number of times the letter *e* appears in it.

Example:

```
count_e("the quick brown fox jumps over the lazy dog.") == 3
count_e("sally sells sea shells by the sea shore.") == 6
count_e("eeeeeeee") == 8
```

Parameterize: Count the number of a given letter

We don't want to have to write a new function every time we want to count a letter. So let's add a *parameter* (also called an *argument*) to the function to let us pick a different letter each time we call the function.

Example:

```
count("e", "the quick brown fox jumps over the lazy dog.") == 3
count("o", "the quick brown fox jumps over the lazy dog.") == 4
count("e", "sally sells sea shells by the sea shore.") == 6
count("a", "sally sells sea shells by the sea shore.") == 3
```