Practice Problems

There are a lot of questions here, and if it feels like an overwhelming number, start with just a couple from each section that you think would help fill in any knowledge gaps you have.

If you find yourself feeling lost, please ask for help in office hours or tutoring.

Quiz 00 General Practice

1. Trace a memory diagram of the following code listing and answer the following subquestions.

```
1    def total_price(calzones: int, strombolis: int) -> int:
2        """Returns the total price for the order of food, including a ser
3        return calzones_price(calzones=calzones) + strombolis_price(strom
4
5    def calzones_price(calzones: int) -> int:
6        """Returns the price of the given number of calzones."""
7        return 7 * calzones
8
9    def strombolis_price(strombolis: int) -> int:
10       """Returns the price of the given number of strombolis."""
11       return 8 * strombolis
12
13   print(total_price(calzones=4, strombolis=2))
```

1.1. What line(s) do *function definition signatures* appear on?

1.2. What line(s) do *docstrings* appear on?

1.3. What line(s) do *expressions* appear on?

1.4. What line(s) do *function calls* appear on?

1.5. Write a function call to `calzones_price` that would evaluate to `28`.

▶ SHOW SOLUTION

2. Trace a memory diagram of the following code listing and answer the following subquestions.

```
1    """Functions of a circle..."""
2
3
4    def main() -> None:
5        """Entrypoint of Program"""
6        print(circumference(radius=1.0))
7        print(area(radius=1.0))
8        return None
9
10
11   def area(radius: float) -> float:
12       """Calculate area of a circle"""
13       return 3.14 * radius ** 2
14
15
16   def circumference(radius: float) -> float:
17       """Calculate circumference"""
18       return 2 * 3.14 * radius
19
20
21   main()
```

2.1. What line(s) do *function definition signatures* appear on?

2.2. What line(s) do *docstrings* appear on?

2.5. What is the return type of `area`? What is the return type of `main`?

▶ SHOW SOLUTION

3. Trace a memory diagram of the following code listing.

```
1    def big_func(a: int) -> int:
2        return a + 2
3
4    def bigger_func(b: int) -> int:
5        return big_func(a=b) * 2
6
7    def biggest_func(num: int) -> int:
8        return bigger_func(b=num) ** 2
9
10   def main() -> None:
11       print(str(biggest_func(num=110)) + " is a big number!")
12
13   main()
```

▶ SHOW SOLUTION

4. Trace a memory diagram of the following code listing.

```
1   def division(x: int, y: int) -> float:
2       return y / x
3       print(y % x)
4
5   print(division(y=64, x=16))
6
7   print(int(64/16))
```

▶ SHOW SOLUTION

5. Trace a memory diagram of the following code listing.

```
1   def start_end(word: str) -> str:
2       return word[0] + word[len(word)-1]
3
4   start_end(word="kitkat")
5   print(start_end(word="skittles"))
```

▶ SHOW SOLUTION

6. Trace a memory diagram of the following code listing.

```
1   def give_cookies(total_cookies: int, num_students: int) -> int:
2       print("Extra cookies: " + str(total_cookies % num_students))
3       return int((total_cookies - (total_cookies % num_students))/2)
4
5   print("Each student gets " + str(give_cookies(total_cookies=11, num_stude
```

▶ SHOW SOLUTION

7. Trace a memory diagram of the following code listing.

```
1    def get_starting_point(word: str) -> int:
2        return int(len(word) / 3)
3
4    def shift_position(index: int) -> int:
```

```
8         return word[index]
9
10    def main(word: str) -> None:
11        print("The hidden character is: " + extract_character(word=word, ind
12
13    main(word="mystery")
```

▶ SHOW SOLUTION

Contributor(s): Alyssa Lytle, Team 110, Izzi Hinks, Benjamin Eldridge, Viktorya Hunanyan, Kaleb White, Rosey Anim, Lynn Lee, Olivia Xiao