

Lecture 8 - Exercises

8A - For Loops

1. What does the following program print?

```
m = 0
for n in [1.8, 4.2, 1.6, 2.0, 1.88]:
    m += int(n)
print(m)
```

2. What is the output of the following program?

```
s = ""
i = 1
for digit in ["one", "two", "three", "four", "five", "six", "seven", "eight", "nine"]:
    if len(digit) < 5 and i % 2 == 1:
        s = digit + s
    i += 1
print(s)
```

8B - The `range` function

3. What does each of the following `range` calls, converted to a list, evaluate to?

- `list(range(3))`
- `list(range(4, 7))`
- `list(range(4, 0, -1))`

4. How many elements are in the following ranges?

- `range(732)`
- `range(6, 88)`
- `range(4, 2)`
- `range(20, 21)`
- `range(0)`

5. What does the following program print?

```
for x in range(1,4):
    print(x + x * x, end=str(x))
```

6. Consider the following incomplete program:

```
x =
y =
for z in range(x, y):
    print("WWU")
```

What pairs of values could be assigned to the variables `x` and `y` so that the program prints `WWU` 43 times? Select all correct choices.

- `x: 0 y: 44`
- `x: -21 y: 22`
- `x: -21 y: 21`
- `x: -789 y: -746`
- `x: -789 y: 746`
- `x: 1 y: 44`

7. Last time you wrote a program that prompts the user for a positive integer p and prints the powers of 2 up to 2^p . Rewrite the program to use a `for` loop instead of a `while` loop.

Problems

1. What does the following program print?

```
for value in [1, 16, 4]:
    print(value, end=" ")
    value = value * 10
```

Hypothesize an answer, then try it out in Thonny to make sure you're right. Make sure you understand why this loop behaves the way it does.

2. **FizzBuzz** The following problem is a commonly-used interview question for programming jobs: Write a program that prints the numbers from 1 to 100. But for multiples of three print `Fizz` instead of the number and for the multiples of five print `Buzz`. For numbers which are multiples of both three and five print `FizzBuzz`. The first few lines of the output look like this:

```
1
2
Fizz
4
Buzz
Fizz
```

3. Recall this program, which we saw in last lecture's Exercises. Write a program that uses `for` loops to print the same output.

```
i = 0
while i < 4:
    j = 1
    while j < 11:
        print('*', end='')
        j += 1
    i += 1
    print()
```

4. Write a program that prints a month of a calendar, such as the following (April of 2021):

```
>>> %Run calendar.py 30 4
su mo tu we th fr sa
                1  2  3
 4  5  6  7  8  9 10
11 12 13 14 15 16 17
18 19 20 21 22 23 24
25 26 27 28 29 30
```

The program takes two command line arguments. The first is an integer number of days in the month (you can assume this will be 28, 29, 30, or 31). The second is a number representing the day of the week that the 1st of the month falls on, with 0–6 representing Sunday through Saturday. Your program can rely on the number of days being between 28–31 (inclusive) and the day of the week being 0–6 (inclusive).

At this point, you may choose to either move on to the following problem, or to go back to any unfinished Problems from the prior lecture. If solving the problems from last lecture, feel free to use whichever kind of loop seems like the best choice to you.

5. How many elements are in the range `range(a, b, c)`? Assume `a <= b` and `c > 0`, and write your answer in terms of `a`, `b`, and `c`.