Cherry Lee H. Jimenez 02/24/24
BSCS-2B Discrete Structures

1. Generate randomly 20 integers from 0 to 10. Place these in “Input” as the input holder.

```
Input: [10, 2, 3, 1, 7, 3, 2, 3, 7, 8, 2, 1, 8, 2, 9, 4, 3, 9, 9, 8]
```

2. Use the inputs in no.1 to generate the output of f(x), place it in “OutputF”. Use the same inputs for g(x) and place it in “OutputG”.

```
OutputF: [5.0, 1.0, 1.5, 0.5, 3.5, 1.5, 1.0, 1.5, 3.5, 4.0, 1.0, 0.5, 4.0, 1.0, 4.5, 2
    .0, 1.5, 4.5, 4.5, 4.0]
```

```
OutputG: [23, 7, 9, 5, 17, 9, 7, 9, 17, 19, 7, 5, 19, 7, 21, 11, 9, 21, 21, 19]
```

3. Find the output of the sum of f(x) and g(x), and place it in “FadditionG”.

```
FadditionG: [28.0, 8.0, 10.5, 5.5, 20.5, 10.5, 8.0, 10.5, 20.5, 23.0, 8.0, 5.5, 23.0,
    8.0, 25.5, 13.0, 10.5, 25.5, 25.5, 23.0]
```

4. Find the output of the quotient of f(x) and g(x), and place it in “FdivisionG”.

```
FdivisionG: [0.21739130434782608, 0.14285714285714285, 0.16666666666666666, 0.1, 0
    .20588235294117646, 0.16666666666666666, 0.14285714285714285, 0.16666666666666666,
    0.20588235294117646, 0.21052631578947367, 0.14285714285714285, 0.1, 0
    .21052631578947367, 0.14285714285714285, 0.21428571428571427, 0.18181818181818182,
    0.16666666666666666, 0.21428571428571427, 0.21428571428571427, 0
    .21052631578947367]
```

5. Find the output of the composition of (f o g)(x), and place it in “FcomposeG”.

```
FcomposeG: [11.5, 3.5, 4.5, 2.5, 8.5, 4.5, 3.5, 4.5, 8.5, 9.5, 3.5, 2.5, 9.5, 3.5, 10
    .5, 5.5, 4.5, 10.5, 10.5, 9.5]
```

Sample Code:

```python
import random

input = []

for i in range(20):
    x = random.randint(0, 10)
    input.append(x)

outputs_f = [x / 2 for x in input]
outputs_g = [2 * x + 3 for x in input]

f_addition_g = [f + g for f, g in zip(outputs_f, outputs_g)]

f_division_g = []
for f, g in zip(outputs_f, outputs_g):
    if g != 0:
        f_division_g.append(f / g)
    else:
        f_division_g.append("Undefined")

f_compose_g = [0.5 * (2 * x + 3) for x in input]

# Printing the results
print("Input:", input)
print("OutputF:", outputs_f)
print("OutputG:", outputs_g)
print("FadditionG:", f_addition_g)
print("FdivisionG:", f_division_g)
print("FcomposeG:", f_compose_g)
```