

## 1 Functions

### Questions

- 1.1 Determine what the Python interpreter will output given the following lines of code.

```
>>> from operator import add, mul
>>> mul(add(5, 6), 8)
```

88

```
>>> print('x')
```

x

```
>>> y = print('x')
```

x

```
>>> print(y)
```

None

```
>>> print(add(4, 2), print('a'))
```

a

6 None

- 1.2 Determine what the Python interpreter will output given the following lines of code.

```
>>> def foo(x):
    print(x)
    return x + 1
```

```
>>> def bar(y, x):
    print(x - y)
```

```
>>> foo(3)
```

3

4

```
>>> bar(3)
```

Error

```
>>> bar(6, 1)
```

-5

```
>>> bar(foo(10), 11)
```

10

0

## 2 Control

### Questions

2.1 Which numbers will be printed after executing the following code?

```
n = 0
if n:
    print(1)
elif n < 2:
    print(2)
else:
    print(3)
print(4)
```

2

4

2.2 WWPD (What would Python Display) after evaluating each of the following expressions?

```
>>> 0 and 1 / 0
```

0

```
>>> 6 or 1 or "a" or 1 / 0
```

6

```
>>> 6 and 1 and "a" and 1 / 0
```

Error

```
>>> print(print(4) and 2)
```

4

None

```
>>> not True and print("a")
```

False

- 2.3 Define a function, `count_digits`, which takes in an integer, `n`, and counts the number of digits in that number.

```
def count_digits(n):
    ...

>>> count_digits(4)
1
>>> count_digits(12345678)
8
>>> count_digits(0)
0
...

count = 0
while n > 0:
    count += 1
    n = n//10
return count
```

- 2.4 Define a function, `count_matches`, which takes in two integers `n` and `m`, and counts the number of digits that match.

```
def count_matches(n, m):
    ...

>>> count_matches(10, 30)
1
>>> count_matches(12345, 23456)
0
>>> count_matches(121212, 123123)
2
>>> count_matches(111, 11) # only one's place matches
2
>>> count_matches(101, 10) # no place matches
0
...

matches = 0
while n > 0 and m > 0:
    if n % 10 == m % 10:
        matches += 1
    n, m = n // 10, m // 10
return matches
```

## 3 Environment Diagrams

### Questions

- 3.1 Draw the environment diagram for evaluating the following code

```
def f(x):  
    return y + x  
y = 10  
f(8)
```

**Solution:** <https://goo.gl/rZnzaM>

- 3.2 Draw the environment diagram for evaluating the following code

```
def dessef(a, b):  
    c = a + b  
    b = b + 1
```

```
b = 6  
dessef(b, 4)
```

**Solution:** <https://goo.gl/4m3NRD>

3.3 Draw the environment diagram for evaluating the following code

```
def foo(x, y):  
    foo = bar  
    return foo(bar(x, x), y)
```

```
def bar(z, x):  
    return z + y
```

```
y = 5  
foo(1, 2)
```

Solution: <https://goo.gl/7Kcx6n>

3.4 Draw the environment diagram for evaluating the following code

```
def spain(japan, iran):  
    def world(cup, egypt):  
        return japan-poland  
    return iran(world(iran, poland))
```

```
def saudi(arabia):  
    return japan + 3
```

```
japan, poland = 3, 7  
spain(poland+1, saudi)
```

Solution: <https://goo.gl/iddW49>

3.5 Draw the environment diagram for evaluating the following code

```
cap = 9  
hulk = 3
```

```
def marvel(cap, thor, avengers):  
    marvel = avengers  
    iron = hulk + cap  
    if thor > cap:  
        def marvel(cap, thor, avengers):  
            return iron  
    else:  
        iron = hulk  
    return marvel(thor, cap, marvel)
```

```
def iron(man):  
    hulk = cap - 1  
    return hulk
```

```
marvel(cap, iron(3), marvel)
```

**Solution:** <https://goo.gl/sofcq2>