CS 61A Functions, Control and Environments
Spring 2019 Guerrilla Section 0: February 2, 2019

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)


>>> print('x')


>>> y = print('x')


>>> print(y)


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

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)


>>> bar(3)


>>> bar(6, 1)


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

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)
```

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

```
>>> 0 and 1 / 0

>>> 6 or 1 or a or 1 / 0

>>> 6 and 1 and a and 1 / 0

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

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

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
```

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 ones place matches
    2
    >>> count_matches(101, 10) # no place matches
    0
```

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)
```

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)
```

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)
```

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)
```

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)
```