CS 61A
Summer 2017

Environment Diagram Party

July 9, 2017

1 WWPD? 61A, Assemble!

```
avengers = 6

def vision(avengers):
    print(avengers)
    return avengers + 1

def hawkeye(thor, hulk):
    love = lambda black_widow: add(black_widow, hulk)
    return thor(love)

def hammer(worthy, stone):
    if worthy(stone) < stone:
        return stone
    elif worthy(stone) > stone:
        return -stone
    return 0

capt = lambda iron_man: iron_man(avengers)
```

Expression	**Interactive Output**
`capt(vision)`	6 7
`print(print(1), vision(2))`	1 2 None 3
`hawkeye(hammer, 3)`	Error
`hawkeye(capt, 3)`	9
`hammer(lambda ultron: ultron, -1)`	0
`hammer(vision, avengers)`	6 6 -6

2 Environment Diagrams **Solved**

A complete answer will:

- Add all missing names and parent annotations to all local frames.
- Add all missing values created or referenced during execution.
- Show the return value for each local frame.

Draw the environment diagram that results from executing the code below until the entire program is finished or an error occurs.

2.1 No "rst" for the Weary

```
def r(s, t):
    while s(a) > b:
        def s(a):
            return t(b)
    return b + 2

def t(t):
    return t - 10

a, b = -10, 2
weary = r(abs, t)
```

https://goo.gl/73BSvB

2.2 Return of the Horse Mask

Solved

```
def bo(jack):
    return lambda mask: horse

def horse(mask):
    horse = mask
    def mask(horse):
        return horse
    return horse(mask)

jack, mask = horse, bo

hollywoo = bo(jack)(horse)(mask)
```

https://goo.gl/yhcHuC

2.3 How Do You Like Your Eggs?

```
def scramble(egg):
    return [egg, over(egg)]

def over(easy):
    easy[1] = [[easy], 2]
    return list(easy[1])

egg = scramble([12, 24])
```

https://goo.gl/nia1NS

2.4 Lazy

```
def lazy(n):
    return lambda k: (n if k == 0 else lazy(n + 1))
v = lazy(4)(1)(0)
```

https://goo.gl/TWFJZc

2.5 Snow Day

```
def snow(snow, x):
    if snow(x, x) == x:
        def x(x):
            return 32
        return x(x)
    else:
        return snow(snow, x)

def flake(x, y):
    return y + x - 1

griffin = snow(flake, 1)
```

https://goo.gl/thDVCx

2.6 Environmentalists

```
def snow(x):
    def ice(x):
        if x == 0:
            return 1
        return 2 + rain(ice, x)

    def rain(g, h):
        return 3 + g(h - x)
    return ice(x)
snow(4)
```

https://goo.gl/Ld4hSm

2.7 Instructor Appreciation

```
def kevin(a, b):
    x = 42
    y = stan(a + b, x)
    return x - y

def stan(a, b):
    a, b = a + x, a - y
    return a // b

x, y = 1, 2
kevin(x, y)
```

https://goo.gl/PRiCpG