Midterm 1 (38+ A+ Question)

1. What Would Python Display?

```
cat = 3
def dog():
    return cat
def hog():
    cat = 4
    print(cat, dog())
cat = 5
```

(a): `dog(cat)`

`Error`

(b): `dog()`

`3` **5**

(c): `print((lambda dog: print(dog()))(lambda: 6))`

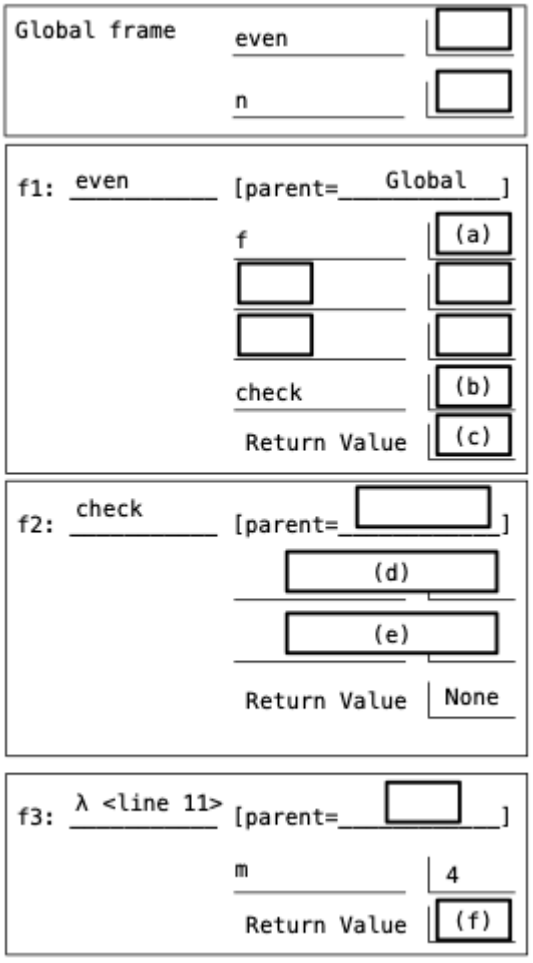
```
6
None
```

(d): `hog()`

`4 5`

---

2. An Odd Implementation




```
 1: def even(f, n):
 2:     "Return whether f(n) is even."
 3:     even = False
 4:     def check(h):
 5:         if h(n) % 2 == 0:
 6:             even = True
 7:     check(f)
 8:     return even
 9:
10: n = 2
11: even(lambda m: m + n, 4)
```

(a): Fill in blank (a)

`func lambda(m) <line 11> [parent=Global]`

(b): Fill in blank (b)

`func check(h) [parent = f1]`

(c): Fill in blank (c)

`False`

(d): Fill in blank (d)

`h is bound to a lambda function.`

(e): Fill in blank (e)

`even is bound to True`

(f): Fill in blank (f)

`6`

(g): What problems are there in the even function's implementation? Select all that apply.

`It always returns False regardless of the value of f(n).`

---

3. In Your Prime

(a): Smallest_gap

- (i) `q - p < k`
- (ii) `q, q+1`
- (iii) `not is_prime(q)`

(b): Plus_one

- (i) `is_prime(max(a, b))`
- (ii) `min`
- (iii) `lambda x, y: (x + y) / 2`

---

4. Choose Wisely

(a): Only

- (i) `t(d)`
- (ii) `d`
- (iii) `while keep`
- (iv) `keep, n = keep // 10, n * 10 + keep % 10`

(b): Every

- (i) `t(n % 10) == False`
- (ii) `return False`
- (iii) `True`
- (iv) `digit`

(c): Even_odd (A+ Question)

Implement `even_odd`, which takes a positive integer `n` that has both even and odd digits. It returns `True` if all of the odd digits of `n` are larger than the last (right-most) even digit of `n`. Otherwise, it returns `False`. You may call `only` and `every`. You may not use `str` or `repr` or `[` or `]` or `for`.

```
def even_odd(n):
    return every(lambda d: d > (only(n, lambda d: d % 2 == 0) % 10))(only(n,
lambda d: d % 2 == 1))
```