https://bit.ly/2S60VPp questions :)

1. What does the following program print?

```
def foo(n, m):
      if (n == 0):
            return m
      else:
            return foo(n - 1, n + m)
x = foo(4, 7)
print(x)
```

   a. 13
   b. 5
   c. 17
   d. Nothing, this program will run forever
   e. *Bruh..*

2. What does the following program print?

```
def blah(b, l):
      ```(int, list) → list```
      if (b == 0):
            return l
      else:
            boop = b%2
            res = b//2
            l.append(boop)
            blah(res, l)
l = []
blah(6, l)
l.reverse()
print(l)
```

   a. [1, 3, 6]
   b. [6, 3, 1]
   c. 011
   d. [0, 1, 1]
   e. [1, 1, 0]
```

3. What does the following program return?

```
def ouf(n, m):
      if (m == 0):
            return n
      return ouf(m, n%m)

val = ouf(20, 12)
print(val)
```

   a. 0
   b. 13
   c. 2
   d. 4

4. How many function calls are done by the previous program (excluding the global frame)?
   a. 3
   b. 4
   c. 5
   d. 6

5. How many function calls are done by the following program (excluding the global frame)?

```
def ouf(n, m):
      if (m == 0):
            return n
      return ouf(m, n%m)

val = ouf(12, 20)
print(val)
```

   a. 3
   b. 4
   c. 5
   d. 6

6. How many times is line 9 run in the following program? (Give the Big-Oh of this program)

```
1.    def foop(A):
2.          ```(list) -> int
3.          A is an unsorted list```
4.          x = 0
5.          n = len(A)
6.          y = 1
7.          bee = abs(A[x] - A[y])
8.          for i in range(n):
9.                for j in range(i + 1, n):
10.                     wasp = abs(A[i]- A[j])
11.                     if (wasp < bee):
12.                          bee = wasp
13.                          x = i
14.                          y = j
15.         l = [x, y]
16.         return bee
```

   a. O(1)
   b. O(n)
   c. $O(n*\log_2 n)$
   d. $O(n^2)$
   e. Wtf Python!?

7. What does the previous program do?
   a. Sorts the list in increasing order
   b. Sorts the list in decreasing order
   c. Returns a list containing the index of the two smallest values in A
   d. Returns the smallest difference between the numbers in A
   e. Bruh 2.0

8. How many times is line 8 executed in the following program? (Give the Big-Oh of this program)

```
1.    def foop(A, t):
2.          ```(list, int) -> int
3.          A is a sorted list, t is the length of A```
4.          x = 0
5.          y = 1
6.          bee = abs(A[x] - A[y])
7.          for i in range(t):
8.                      wasp = abs(A[i]- A[j])
9.                      if (wasp < bee):
10.                           bee = wasp
11.                           x = i
12.                           y = j
13.         l = [x, y]
14.         return bee
```

   a. O(1)
   b. O(n)
   c. O(n*$\log_2$n)
   d. O(n^2)
   e. *Python! I said WTF!?*

9. Fill in the blank

```
1. def fact(num):
2. if num == 0:
3.         return 1
4.    else:
5.         return _____________________
```

   a. num*fact(num-1)
   b. (num-1)*(num-2)
   c. num*(num-1)
   d. fact(num)*fact(num-1)

10. Give the Big-Oh of this program

```
1. def alg(A,n):
2.    tot = 1
3.    for i in range(n-1):
4.      for j in range (100):
5.            for k in range (i*i):
6.                  tot = tot+A[i][j] * k
7.     return tot;
```

a. O(1)
b. O(n)
c. $O(n^2)$
d. $O(n^3)$
e. $O(n^4)$