

15-112 F21 Practice Quiz 2

25 minutes

Note: This practice quiz was written by TAs. It may not reflect the exact difficulty or length of the actual quiz.

(But it does reflect the pizza preferences of the TAs who created it.)

1. Short Answer & Multiple Choice

Suppose you're solving a problem in which you want to find how many digits of a number of unspecified length are equal to 1. What kind of loop would you use?

What is the difference between break and continue?

- A. break ends the current iteration, but continue ends the entire loop
- B. continue skips the rest of the current iteration, and break ends the entire loop
- C. continue can only be used in loops while break ends a function
- D. continue continues onto the next line but break ends the loop

What sequence of numbers will the following line of code iterate over?

```
for i in range(20, 4, -4):
```

Which of the following is **not** a valid Python debugging strategy?

- A. Using print statements to print out relevant variables
- B. Using input to pause the program at certain points
- C. Using gdb to step through your code line by line
- D. Using locals to see all the variables currently in scope

2. **Code Tracing:** Indicate what these print. Place your answers (and nothing else) in the boxes below the code.

```
def ct1(n, d, k):
    print("pineapple on pizza?", end = " ")
    for i in range(0, 12, 3):
        for j in range(0, i):
            # Be careful of order of operations!
            # Hint: Can the condition be True when i is nonzero?
            if(i + j % 2 == 0):
                n += i
                d += j
            else:
                k += 1
        if(i % 2 == 0):
            print("heck yeah!", d + k, end=" ")
        elif(i == 12): # What is the greatest value i ever attains?
            print("ew no!", n, end=" ")
    return (n + k + d)
print(ct1(0, 0, 0))
```

```
def ct2(i, j, k):
    while True:
        if(i * j == 24): return i + j
        if(i % 2 == 0):
            print("hotel?", k)
            k += 1
        if(j % 3 == 0):
            print("trivago", k)
            k += 2
        i //= 2
        j //= 2
    print(ct2(70, 30, 0))
```

```
def ct3(n):  
    m = 0  
    i = 0  
    while n > 0:  
        d = n % 10  
        n //= 10  
        if d % 2 == 0:  
            continue  
        print(f"{i} -> {m}") # hmm, this looks familiar...  
        m = m * 10 + d  
        i += 1  
        if m > 100:  
            break  
    return m  
print(ct3(15112))
```



3. Free Response: nthNearlyNarcissistic(x, r)

We define a narcissistic number as a non-negative integer x with the property that x is equal to the sum of each of its digits taken to the power of the number of digits in x . For example,

$$371 = 3^3 + 7^3 + 1^3$$

so 371 is narcissistic.

With this in mind, write the function `nthNearlyNarcissistic(n, r)` that takes in an integer n and returns the n th positive integer that is at most r away (in either direction) from a narcissistic number. Note that we **are** counting numbers that are exactly r away from a narcissistic number.

Below are some examples:

```
# Tests for isNarcissistic
assert(isNarcissistic(0) == True) # 0**1 == 0
assert(isNarcissistic(7) == True) # 7**1 == 7
assert(isNarcissistic(1634) == True) # 1**4 + 6**4 + 3**4 + 4**4 == 1634
assert(isNarcissistic(92727) == True) # 9**5 + 2**5 + 7**5 + 2**5 + 7**5
                                     # == 92727
assert(isNarcissistic(112) == False) # 1**3 + 1**3 + 2**3 = 10 != 112

# Tests for nthNarcissistic
assert(nthNearlyNarcissistic(0, 0) == 0)
assert(nthNearlyNarcissistic(0, 42) == 0)
assert(nthNearlyNarcissistic(14, 0) == 1634)
assert(nthNearlyNarcissistic(14, 4) == 149) # 149 is 4 away from 153, a
                                     # narcissistic number
assert(nthNearlyNarcissistic(14, 5) == 14) # 14 is 5 away from 9
assert(nthNearlyNarcissistic(420, 70) == 1585) # 1585 is 49 away from 1634
```

Hint: You may find it helpful to define a helper function that determines whether a number is within r of a narcissistic number.

You may write your `nthNearlyNarcissistic(x, r)` function in the space below.