

15-112 Spring 2019 Quiz 2

Up to 20 minutes. No calculators, no notes, no books, no computers. Show your work!

Do not use strings, lists, dictionaries, try/except, or recursion on this quiz.

1. (10 points) **Debugging:** Consider the following function designed to sum the digits of a number. It contains a bug. Find and fix the bug by changing exactly one line of code. (Write your change on the code itself.) Then, describe what the bug is in the box next to the code.

```
def digitSum(n):

    while n > 0:

        sum += (n%10)

        n //= 10

    return sum
```

2. (20 points) **Code Tracing:** Indicate what the following program prints. Place your answer (and nothing else) in the box next to the code.

```
def ct1(a, b, c):
    for i in range(2,b,c):
        n = a
        while n > 0:
            print(i, n)
            n -= c
        a -= 1

print(ct1(7,6,3))
```

3. (20 points) **Reasoning Over Code:** Find an argument (the value of n) for the following function that makes it return True. Place your answer (and nothing else) in the box below the code:

```
def argh(n, b):
    c = 0
    while n > 0:
        c += n%10
        n //= b
    return c

def blarg(n, a, b):
    while n > 0:
        d = n%10
        if d == 0 or d%2 == a:
            return False
        n //= b
    return True

def rocl(n):
    return argh(n,100) == argh(n//10,100) \
        and blarg(n,1,100) == blarg(n//10,0,100) == True \
        and n > 999 and n < 9999
```

4. (50 points) **Free Response:** We'll say that an integer is a corgish number (coined term) if all the digits are odd and the first and last digit are both 7.

Write the function `nthCorgishNumber(n)` which takes a non-negative integer `n` and returns the `n`th corgish number. `nthCorgishNumber(0)` should return 7. The first several corgish numbers are: 7, 77, 717, 737, 757, 777, 797, 7117, 7137, 7157, 7177, 7197, 7317, and 7337.

Hint: You should probably write a helper function.