

Name: _____ Section: ___ Andrew Id: _____

15-112 Fall 2016 Quiz 2a

* Up to 25 minutes. No calculators, no notes, no books, no computers. * Show your work!

* No strings, lists, or recursion

1. **Code Tracing** [24 pts]: Indicate what these print. Place your answers (and nothing else) in the boxes beside the code.

```
def ct1(x, y):
    for z in range(y, x):
        print(42, z, end=' ')
    for z in range(x, y):
        if (z < y//2):
            if (z%2 == 0): print (2, z, end=' ')
            elif (z%5 == 0): print(5, z, end=' ')
        elif (x+y+z == 27):
            print(27, z, end=' ')
    w = 0
    for z in range(x, 10*x, x):
        if (z < 5*x): continue
        elif (z >= 7*x): return w
        w += z
    return 99
print(ct1(3, 13)) # prints 7 values
```

```
def ct2(x):
    counter = 0
    target = 2 # does not start at 0
    while (target < 4):
        if ((x + counter//2) == target):
            print(counter, target, end=' ')
            target += 1
        counter += 1
    return 10*counter+target
print(ct2(2)) # prints 5 values
```

2. **Reasoning Over Code** [12 pts]:

Find an argument for the following function that makes it return True. Place your answer (and nothing else) in the box below the code:

```
def rc1(m):
    if ((not isinstance(m, int)) or (m < 0)): return False
    x = 1
    while (x < m//2): x += 10
    return (x + m == 60)
```

m =

3. **Short Answers** [20 pts] Be very brief.

- a. What is `carrylessAdd(329, 817)`?
- b. What is `nthEmirp(0)`?
- c. If $f(x) = \text{abs}(x)$, what will `findZeroWithBisection(f, -1, 1, .00001)` return?
- d. To show that 42 is not happy, we first convert 42 to N, and then N to 4, which is unhappy. What value is N?

4. **Free Response 1: sameOddestDigit(m, n)** [44 pts]

Say that the 'oddest' digit of a number (a coined term) is the odd digit that occurs the most in that number, with ties going to the larger digit, or 0 if the number has no odd digits. For example, the oddest digit of 123454321 is 3.

Similarly, the oddest digit of -123123 is also 3. With this in mind, write the function `sameOddestDigit(m, n)` that takes two possibly-negative int values and returns True if they have the same oddest digit and False otherwise. So:

```
sameOddestDigit(123454321, -123123) == True   sameOddestDigit(123454321, -123122) == False
sameOddestDigit(2468, 24689) == False         sameOddestDigit(2468, 20000) == True
```

Hint: You may want to write one or two well-chosen helper functions. Remember not to use strings.

5. **Bonus/Optional: Code Tracing** [2.5 pts] What will this print? Place your answer in the box.

```
def bonus(x):
    (a,b,c) = (x%10,x//10%10,x//100%10); x = 100*(b-1)+10*(a-1)+(c-1)
    return x if (min(a,b,c)==0) else bonus(x)
print(bonus(873))
```