Name:_____ Section:___ Andrew Id: _____

15-112 Spring 2017 Quiz 1a

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

*** No strings, lists, loops, or recursion**

1. **Code Tracing** [30 pts]:Indicate what these print. Place your answers (and nothing else) in the boxes below the code.

```
def ct1(x, y, z):
  print(x + y ** 2 - z * 2)
  print(100*int(y/x) + y/x)
  print(100*math.ceil(y/x) + y//x)
  y %= x + z
  return y
print(ct1(2, 5, 7)) # hint: prints 4 values
```

```
def f(x): return 2*x-1
def g(x): return max(x%10, x//10%10)
def h(x):
  if (x > 0): return x + g(f(x))
  else: return f(min(3, g(abs(x))))
def ct2(x):
  print(h(x-5))
  print(h(x+5))
print(ct2(3)) # hint: prints 3 values
```

2. **Reasoning Over Code** [10 pts]:
   Find an argument for the following function that makes it return True. Place your answers (and nothing else) in the boxes below the code:

```
def rc1(n):
    if ((not isinstance(n, int)) or (n < 100) or (n > 999)):
      return False
    a = n%10
    b = n%100
    c = n//100
    return ((a == b) and (a == c) and (abs(n - 600) == a))
```

n =

3. **Short Answers** [20 pts]

   a. Very briefly, why does the linter disallow using round(x)?

   b. What is the smallest positive integer x such that ((x >= 50) and (x % 20)//10 == 0)?

   c. Write one line of code that will produce a runtime error (and not a syntax error nor a logical error).

   d. If x is an int, and (x % 7 > x), what else must be true about x?

Name:_____ Section:___ Andrew Id: _____

4. **Free Response 1: isTinyPal(n)** [40 pts]

Background: we will say that an integer is a palindrome if it the same frontwards as backwards, ignoring its sign and ignoring leading 0's. So 12421 and -80008 are both palindromes. Further, we will say that an integer is a tiny palindrome, or a "tinyPal", if it is a palindrome and has no more than 3 total digits, again ignoring leading 0's. With this in mind, write the function isTinyPal(n) that takes any Python value and returns True if it is a tiny integer palindrome, and False otherwise. So isTinyPal(-808) returns True, but these all return False (without crashing):

```
isTinyPal(80009) returns False   # not a palindrome
isTinyPal(80008) returns False   # not tiny
isTinyPal(8000.8) returns False  # not an int
isTinyPal("80008") returns False # not an int
```

Hint: since we cannot use loops or strings yet, there is no way to check if an arbitrary int is a palindrome. But we are heavily restricting ints here to be tiny, with 3 or fewer digits, and so we can do it just with what we've covered so far in this course.

Quiz continues on next page!

5. **Bonus/Optional: Code Tracing** [5 pts]
   Indicate what this prints. Place your answer (and nothing else) in the box below the code):

```
def bonusCt1(n):
  global min, max, pow
  min = max; max = pow; pow = min
  return max(2*n, min(3*n, pow(n, n+1)))
print(bonusCt1(2))
```

```
def bonusCt2(x, y):
  return (x+bonusCt2(x-1,y-1) if (y > 0) else
          (2+bonusCt2(x-1,y) if (x > 0) else 42))
print(bonusCt2(7, 4))
```