Name:_____________________________________ Section:___ Andrew Id: ______________________ a

15-112 Spring 2020 Quiz 1
Up to 15 minutes. No calculators, no notes, no books, no other paper, no computers.
No strings, lists, string or list indexing, loops, or recursion
You may call almostEqual(x, y) and roundHalfUp(d) without writing them. Write everything else!

1. True or False [18pts]

Indicate whether each of the following is True or False by **completely** filling in one bubble for each.

○True ○False Runtime errors are raised immediately before your code begins running.

○True ○False If code compiles and runs but gives us an incorrect answer, it has a short circuit error.

○True ○False `isinstance("42",str)` returns `True`.

○True ○False If `x = 4 / 2` then x is a float.

○True ○False The expression `("123" * 3)` returns `"123123123"`.

○True ○False Floats and integers always return False when compared with ==,
for example: `10 * (0.1 + 0.1 + 0.1) == 3`

2. Code Tracing 1 [15pts]

Indicate what the following code prints. Place your answer (and nothing else) in the box to the right.

```
def ct1(x):
    print(x + 10)
    x = x + 25 // 10
    print(x)
    x -= 3
    print(x)
    return x == 0

ct1(10)
```

3. Code Tracing 2 [15pts]

Indicate what the following code prints. Place your answer (and nothing else) in the box to the right.

```
def ct2(x, y, z):
    print(x/y + x//y + int(x/y))
    print(y**z + y%z)
    a = int(x) / int(y)
    return isinstance(a, int)

print(ct2(6, 4, 3))
```

4. Reasoning over Code 1 [10pts]

Find an argument for the function rc1(n) that makes it return True. Place your answer (and nothing else) in the box to the right. **(Hint: Watch out for crashes!)**

n =

```
def rc1(n):
    if (n*10)//1 == 112:
        n*=100
        if n%10<5:
            return False
    else:
        if n%10<5 and n//0 == 0:
            return True
        n=0
    return n>1
```

5. Free Response: `splitPower(x, n)` [42pts]

Write the function `splitPower(x, n)` where x and n are integers, and n is non-negative. This function removes the n rightmost digits of x, leaving `0` or more leftmost digits. The function then returns the number formed by the leftmost digits raised to the power of the rightmost digits. For example, `splitPower(23, 1)` returns 8, because `2**3 == 8`. Likewise, `splitPower(123, 2)` returns 1, because `1**23 == 1`. Note that `splitPower(-902, 2)` returns 81 because `(-9)**2 == 81`.
Be careful with negative numbers!

6. Bonus CT1 (This problem is optional!) [3pts]

Indicate what the following code prints. Place your answer (and nothing else) in the box to the right.

```
def   bonusCt1(n):
    def f(n): return 1 - 1/n
    return max(n, f(n), f(f(n)), f(f(f(n))), f(f(f(f(n)))),
               f(f(f(f(f(n))))), f(f(f(f(f(f(n)))))))
print(0.5 + bonusCt1(0.5))
```