Name: _____ Recitation: _____ Andrew Id: _____

15-112 Fall 2018 Quiz 2

Up to 20 minutes. No calculators, no notes, no books, no computers, no other paper. Show your work!
Do not use recursion, lists, or string indexing on this quiz. Do not convert ints or floats to strings, and do not convert strings to ints or floats!

Answer both sides of this quiz!

1. (5 points) **Short Answer:** Which type of loop (`for` or `while`) is **typically** used when an indeterminate or variable number of iterations may be needed (i.e. which loop type is typically used when the number of required iterations is unknown when the loop begins)?

2. (10 points) **Debugging:** The piece of code shown below is supposed to print a sequence of numbers and return the final value, but it has SEVERAL bugs. When we run this code, what error will Python raise FIRST? Circle the part of the code that the first error will point to. Then, identify which of the three basic error types this is, and write the buggy line correctly. Place your answer (and nothing else) in the box to the right of the code. You only need to do this for the FIRST error!

```
1 def errorFun(x):
2     z = 1
3     for i in range(0,4):
4         if i > 2
5             z = 0
6         x += x // z
7         z += 1
8         print(x)
9     return x
10
11 assert(errorFun(1)==4)
```

3. (20 points) **Code Tracing:** Indicate what the following program prints. Place your answer (and nothing else) in the box to the right of the code.

```
def ctFun(xMax,yMax):
    for x in range(0,xMax):
        for y in range(yMax,x-1,-1):
            if x==y:
                print(y+x)
            else:
                print(y+x,end=" ")
    return x+y

ctFun(3,4)
```

4. (25 points) **Reasoning Over Code**: Find an argument for `rocFun(x)` that makes it return `True`. Place your answer (and nothing else) in the box to the right of the code.

```
def foo(x):
    c=0
    while x>0:
        c+=x%10
        x=x//10
    return c

def bar(x):
    c=0
    while x>0:
        c+=1
        x=x//10
    return c**2

def rocFun(x):
    y=1
    while x>y:
        z=x%y
        y=y*10
        assert(foo(z)==bar(z))
    return bar(x)==25
```

5. (40 points) **Free Response:** Write the function `nthSnarf(n)` which returns the nth Snarf number (a coined term). A Snarf is any positive integer with at least two digits where no even digit touches another even digit, no odd digit touches another odd digit, and the digits are strictly decreasing from left to right. For example, 32, 3210, 652, and 94 are Snarfs, 33, 325, 532, 8341, and -94 are not. The base case is `nthSnarf(0)==10`.
Hint: It may help to write the first few Snarfs first, and then write the helper function `isSnarf(num)`! For this problem, do not use strings of ANY kind, indexing, lists, recursion, or floats!