Name: _____ Recitation: _____ Andrew Id: _____

15-112 Summer 2018 Quiz 1

Up to 50 minutes. No calculators, scratch paper, notes, books, or computers. Do not use any topics that were not taught on during week1. In particular, do not use lists or recursion. Show your work!

1. (25 points) **Free Response**: Write the function leftOverLetters(letterBank, targetWord) that takes in two strings, a letterBank and targetWord, and returns the number of leftover letters in the letterBank after constructing the targetWord from letters in the letterBank. Constructing the word should be case-insensitive. If you cannot create the targetWord with the given letterBank, return -1.

   For example:

   leftOverLetters("atc", "cat") == 0

   leftOverLetters("12", "112") == -1

   leftOverLetters("abcmude", "CMU") == 4

   leftOverLetters("aaa", "a") == 2

   You may not use lists or recursion on this question.

2. (40 points) **Free Response**: A positive number is considered "evenHeavy" if the sum of the digits in even indexed positions is larger than the sum of the digits in odd indexed positions. We start counting digits from the right side of the number, where the right-most digit is at index 0.

   For example, in the number 123, 3 and 1 are at even indexed positions, with 3 at index 0 and 1 at index 2. There is one digit in an odd indexed position, and that is 2 at index 1. Therefore, 123 would be an "evenHeavy" number because $3+1 > 2$.

   Here are some more examples:

   isEvenHeavy(121212) == True

   isEvenHeavy(212121) == False

   isEvenHeavy(2222) == False

   isEvenHeavy(12321) == True

   Write the function nthEvenHeavy(n) that takes a non-negative integer n and returns the nth "evenHeavy" number. nthEvenHeavy(0) should return 1.

   The first several "evenHeavy" numbers are: 1,2,3,4,5,6,7,8,9,12,13,14,15,16,17,18,19,23,24,25,26,27...

   You may not use strings, lists, or recursion on this question. Therefore, you cannot use str.

3. (5 points) **Short Answer**

   (a) What is the length of the string: "I<3\n112"?

   (b) If there is no return statement, what does a function return?

   (c) What is -7//3?

   (d) Name a TA on the course staff.

   (e) How can using try and except lead to difficulties when debugging?

4. (20 points) **Code Tracing**

   Indicate what the following programs print. Place your answers (and nothing else) in the box under the code. Show your work anywhere outside the box.

   (a) (8 points) CT1

```
def f(x):
    if(x % 2 == 0):
        return 4*x + 2
    else:
        return 3*x + 1

def g(x): return max(f(x//3), x)

def ct1(x):
    print(g(x) * 2)
    x %= 3
    print(x)
    print((f(x+2)%10)*10)
    x += 7
    print(x)
    return f(g(x)%4)

print(ct1(5) + 5)
```

(b) (12 points) CT2

```
def ct2(n, m):
    for i in range(1, 2*n, n):
        for j in range(-m, -1, m//3):
            if(i % 4 == j):
                print("mod", i, j)
            if(i == j):
                print("equal", i, j)
            elif(j > i):
                print("greater", i, j)
            else:
                break

ct2(2, -5)
```

5. (10 points) **Reasoning Over Code**: Find an argument (the value of s) for roc that makes it return True. Place your answer (and nothing else) in the box. Show your work anywhere outside the box.

```
def roc(s):
    t = "1a1b2c"
    assert(isinstance(s, str) and len(s) == len(t))
    length = len(s)
    result = ""
    for index in range(length):
        c1 = t[index]
        c2 = s[length - 1 - index]
        if(c1 == c2):
            result += c1
        elif(c1 in s):
            result = result.replace(c1, "")
    return result == "112"
```