Name:_____ Section:___ Andrew Id: _____

15-112 Fall 2014 Quiz 4

*** 17.5 minutes. No calculators, no notes, no books, no computers.**
*** You may not discuss any portion of this quiz with anyone until after 5pm today.**
*** No lists or recursion! * SHOW YOUR WORK, CIRCLE YOUR ANSWERS.**

1. **Code Tracing** [20 pts; 10 pts each]: Indicate what this will print:

```
def t1(s, t):
    for c in s:
        if (c.upper() not in "NO!!!"):
            i = t.find(c)
            print i, s[i], t[i],

t1("net", "two")
```

```
def t2(s):
    result = ""
    d = ord("a")
    for c in s.lower():
        if (c.isalpha() and (ord(c) >= d)):
            result += str(ord(c) - d)
            d += 1
    return result

print t2("Be a CA?!?")
```

2. **Reasoning Over Code [10 pts]**
Find arguments for the following function that make it return True. You only need one set of arguments for the function, even if there are multiple correct answers.

```
def f(s, t, n):
    q = ""
    for i in xrange(len(s)):
        q += t[(i+n)%len(t)]
    print q
    return ((len(s) == 2*len(t)) and
            (not s.startswith(t)) and
            (q == s))
```

3. **Free Response: wordWrap** [70 pts]
Write the function wordWrap(text, width) that takes a string of text (containing only lowercase letters or spaces) and a positive integer width, and returns a possibly-multiline string that matches the original string, only with line wrapping at the given width. So wordWrap("abc", 3) just returns "abc", but wordWrap("abc",2) returns a 2-line string, with "ab" on the first line and "c" on the second line. After you complete word wrapping in this way, only then: All spaces at the start and end of each resulting line should be removed, and then all remaining spaces should be converted to dashes ("-"), so they can be easily seen in the resulting string. Here are some test cases for you:

```
    assert(wordWrap("abcdefghij", 4) == """\
abcd
efgh
ij""")
    assert(wordWrap("a b c de fg", 4) == """\
a-b
c-de
fg""")
```

4. **Bonus/Optional**: What will this print? [5 pts]

```
import string
def bonusTrace(i):
    # hint: remember that ord("A") is 65, ord("a") is 97, and ord("0") is 48
    pattern = result = ""
    for c in string.ascii_uppercase: pattern += str(ord(c))
    for x in xrange(100*i, 100*(i+1)):
        if (str(x) in pattern): result += chr(x%100)
    return result
print bonusTrace(4)
```