CIS 210
Winter 2014 Midterm Exam

Write your name at the top of **each page** before you begin. [5 points]

1. [5 points] What does `q1( )` print?

```
def q1():
    li = [1, 2, 3, 4, 5, 6 ]
    ev = 0
    od = 0
    for n in li:
        if n % 2 == 0:
            ev += n
        else:
            od += n
    print( ev - od )
```

2. [5 points] What does `q2( )` print?

```
def cnt(el, li):
    """I'll surely lose points for this terrible docstring"""
    c = 0
    for i in range(len(li)):
        if li[i] == el:
            c += 1
    return c

def q2():
    ar_x = [ 0, 1, 0, 2, 2, 3, 0 ]
    ar_y = [ "alpha", "beta", "gamma", "delta" ]
    sum = cnt(2, ar_x) + cnt("beta", ar_y)
    print(sum)
```


(score)

3. [5 points] What does `q3( )` print? (Recall that `//` is integer division.)

```
def i_scale(ar, sf):
    for i in range(len(ar)):
        ar[i] = ar[i] // sf
    return

def q3():
    li = [ 4, 5, 6, 7 ]
    i_scale(li, 2)
    sum = 0
    for n in li:
        sum += n
    print(sum)
```

4. [5 points] What does `q4( )` print?

```
def rec(li,i):
    if i >= len(li):
        return 0
    else:
        return li[i] + rec(li, i+1)

def q4():
    ar = [ 1, 2, 3, 4, 5 ]
    t = rec(ar,0)
    print(t)
```

5. [11 points] Complete the function `phone_to_int`, without using the Python built-in function `int( )`. The Python quick reference sheet includes a reminder of how to use a `dict` structure like `DIG_VAL`. It may also be useful to remember that you can build up integers by multiplying and adding, e.g., 10*42 + 7 = 427.

```
DIG_VAL = { "0": 0, "1":1, "2":2, "3":3, "4":4,
            "5":5, "6":6, "7":7, "8":8, "9":9 }

def phone_to_int(ph):
    """
    Convert phone number to integer.
    Args:
       ph: A string representing a phone number. ph may contain
         digits 0-9, spaces, punctuation, and other characters.
    Returns:
       an integer representing just the digits in ph
    Examples:
       phone_to_int("(341) 556-9897") = 3415569897
       phone_to_int("34-45-(442).22") = 344544222
       phone_to_int("000-000-92") = 00000092 = 92
       phone_to_int("there are no digits here") = 0
       phone_to_int("") = 0
    """
    # Your code here
```


(score)

6. [14 points] Finish the function `respace` below, consistent with the docstring.

```
def respace(s):
    """
    Collapse runs of spaces.
    Args:
       s:  a string
    Returns:
       a copy of s, except that every run of 2 or more spaces has been
       replaced by a single space
    Examples (with _ representing a space):
       respace("a__b_c__") = "a_b_c_"
       respace("__a___b") = "_a_b"
       respace("abcde") = "abcde"
       respace("") =  ""
       respace("_____") = "_"
    """
    # Your code here
```