CSE 160 Section 2 Problems

1. Fill in the necessary code to build the list ages

   ages = [] #creates an empty list
   for age in [20, 21, 20, 22, 19, 18, 14, 35]:
       print ages

   The output should be [20, 21, 20, 22, 19, 18, 14, 35]
   (Hint: x.append(2) appends the number 2 to the end of list x)

2. Write a function, over_twenty() to count the number of people over 20 years old in the list ages.

3. Write the output to the following problem:

   grid = [[1, 2, 3], ['a', 'b', 'c'], ['c', 's', 'e'], [1, 4, 0]]
   print grid[0][0]
   print grid[1][2]
   print grid[2][1]
   print grid[3][2]

4. Modify the following code so that it properly adds 5 to everyone's age

   ages = [20, 21, 20, 22, 19, 18, 14, 35]
   for i in ages:
       ages[i] + 5
   print ages

   print ages should now return [25, 26, 25, 27, 24, 23, 19, 40]

5. Write a function that calculates and returns the average of ages. You are not allowed to use python's built-in sum() function. Your function should take in the list ages as a parameter and return the average.
6. Given a function get_height that computes the height of the student passed in, write a new function max_height that finds the maximum height of all the people in the class. Your function should take in a list of student names and return the maximum height. You can assume height is in inches and that the list of all students in the class is class_lst.

get_height(‘nicholas’) will return 75

What is the type of max_height(students)?

Suppose the code was modified to print max_height instead of return max_height, what would be the type of max_height(students)?
1.  
   ```python
   ages = [] #creates an empty list
   for age in [20, 21, 20, 22, 19, 18, 14, 35]:
       ages.append(age)
   print ages
   ```

2.  
   ```python
   def over_twenty(ages):
       total = 0
       for age in ages:
           if age > 20:
               total = total + 1
       return total
   ```

3.  
   ```
   1
   c
   s
   0
   ```

4.  
   ```
   ages = [20, 21, 20, 22, 19, 18, 14, 35]
   for i in range(len(ages)):
       ages[i] = ages[i] + 5
   print ages
   ```

5.  
   ```python
   def avg_age(ages):
       total = 0
       for age in ages:
           total = total + age
       avg = float(total)/len(ages)
       return avg
   ```

6.  
   ```python
   def max_height(class_lst):
       max_height = 0
       for student in class_lst:
           student_height = get_height(student)
           if(student_height > max_height):
               max_height = student_height
       return max_height
   ```

   Type when returning: Int
   Type when printing: None