PROBLEM 1 :  (What is the output? (10 points))

A. (5 pts) What is the output of the following code segment? Write the output to the right. Note that there is only output for the print statements.

Output:

```python
x = 5
y = 3.0
print 2 + x * 4
print x * 4 + 2
print 12 / x
print 10 / y
print x > 2
```

B. (5 pts) What is the output of the following code segment? Write the output to the right. Note that there is only output for the print statements.

Output:

```python
sport = "cross country"
print sport[2]
print sport[2:4]
print sport[sport.find('t'):]"

spices = ['basil', 'dill', 'chickory', 'parsley', 'sage']
print spices[2]
print spices[2:4]
```

PROBLEM 2 :  (Triangles and Dinner - Simple Functions (14 points))

A. (6 pts) An equilateral triangle has all three sides of the same length. The area of an equilateral triangle with $a$ the length of a side is $\frac{\sqrt{3}}{4} a^2$.

Write the function `areaEquilateralTriangle` that has one float parameter `side` representing one side of an equilateral triangle and returns the area of the triangle.
def areaEquilateralTriangle(side):
    
    given one float parameter side representing the length of
    one of the sides,
    returns a float that is the area of the equilateral triangle
    
    
B. (8 pts) Ellen and Oscar want an easy way to decide who will cook dinner each night. They decided that Oscar will cook if it is an odd day and Ellen will cook if it is an even day. But then Oscar realized that many times he will cook two days in a row, on the 31st and the 1st, but that Ellen would never cook two days in a row. They then agreed in addition that if the day was the 31st day of the month then if the month was even, Ellen would cook and if the month was odd, Oscar would cook that day.

Write the function whoseNightToCook that has two int parameters day and month and returns the string 'Ellen' or 'Oscar', the name of the person who should cook dinner that night following the rules above. Assume the arguments are correct. That is, you do not need to know or verify how many days in a month.

def whoseNightToCook(day, month):
    
    returns "Ellen" if an even day, "Oscar" if an odd day except when day is 31, returns "Ellen" if month is even, and "Oscar" if month is odd
PROBLEM 3: (It's a mystery (12 points))

Consider the following mystery function with one parameter animals which is a list of strings.

```python
def mystery(animals):
    ''' animals is a list of strings '''
    x = []
    for w in animals:
        x.append(len(w))
    amount = max(x)
y = [w for w in animals if len(w) == amount]
    return y[0]
```

A. (4 pts) Consider making the call `mystery(animals)` with the value of `animals` below. Answer the following questions about tracing what happens with this call.

```python
animals = ['cat', 'mouse', 'snake', 'chicken', 'fish']
```

A1. What is the value of `x` after line 5 executes?
A2. What is the value of `amount` after line 5 executes?
A3. What is the value of `y` after line 6 executes?
A4. What value is returned from the call `mystery(animals)`?

B. (8 pts) Consider making the call `mystery(zoo)` with the value of `zoo` below. Answer the following questions about tracing what happens with this call.

```python
zoo = ['lion', 'rhino', 'bear', 'zebra']
```

B1. What value is returned from the call `mystery(zoo)`?
B2. Explain in words what `mystery` does.
B3. Rewrite lines 2-5 as one line that includes a list comprehension
B4. In the original code, if line 7 was changed to `return y[-1]`, explain in words what `mystery` would now do.
PROBLEM 4:  (How many teens? How many boomers? (16 points))

A. (8 pts) Write the function getAges which has one parameter data that is a nonempty list of strings in the format 'firstname:lastname:age' and returns a list of ints of the ages from data.

<table>
<thead>
<tr>
<th>call</th>
<th>returns</th>
</tr>
</thead>
<tbody>
<tr>
<td>getAges([&quot;Barack:Obama:50&quot;]</td>
<td>[50]</td>
</tr>
</tbody>
</table>

B. (8 points) Write the function howManyInRange which has three parameters, data that is a list of strings in the format 'firstname:lastname:age', and two int parameters start and end. This function returns the number of people in the age range from start to end including the start and end ages. In writing howManyInRange you may call getAges that you wrote in Part A. Assume getAges works correctly.

<table>
<thead>
<tr>
<th>call</th>
<th>returns</th>
</tr>
</thead>
<tbody>
<tr>
<td>howManyInRange([&quot;A:A:8&quot;, &quot;B:B:3&quot;, &quot;C:C:17&quot;, &quot;D:D:42&quot;, &quot;E:E:20&quot;],20,29)</td>
<td>1</td>
</tr>
<tr>
<td>howManyInRange([&quot;Barack:Obama:50&quot;],30,39)</td>
<td>0</td>
</tr>
</tbody>
</table>

PROBLEM 5:  (Talk like a Pirate (14 points))

There are three simple rules for talking like a pirate.

1. The word 'Hello' (capitalized or not) becomes 'Ahoy' (always capitalized)
2. 'ar' not starting a word becomes 'arrr' (replace only the first occurrence)
3. For any word of length greater than 7 that does not contain 'ar' inside the word, remove all occurrences of lowercase o’s and u’s

Write the function convertWord that takes a word and returns the pirate equivalent of that word following the rules above.

def convertWord(word):
<table>
<thead>
<tr>
<th>call</th>
<th>returns</th>
<th>comment</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>convertWord('yesterday')</code></td>
<td>'yesterday'</td>
<td>no changes</td>
</tr>
<tr>
<td><code>convertWord('boing')</code></td>
<td>'boing'</td>
<td>word too short, no o replaced</td>
</tr>
<tr>
<td><code>convertWord('are')</code></td>
<td>'are'</td>
<td>'ar' starts a word, no change</td>
</tr>
<tr>
<td><code>convertWord('gargargantuan')</code></td>
<td>'garrrgargantuan'</td>
<td>only first 'ar' replaced</td>
</tr>
<tr>
<td><code>convertWord('purposefully')</code></td>
<td>'prpseffly'</td>
<td>o's and u's removed</td>
</tr>
<tr>
<td><code>convertWord('starboard')</code></td>
<td>'starrrboard'</td>
<td>no 'o' removed since 'ar' in word</td>
</tr>
</tbody>
</table>