Point Values: 1-12, 2-15, 3-10, 4-10, 5-15 (Total: 62)

1. For each of the following expressions, determine both the type and the value. One or more of the expressions is incorrectly formed, and in that case, you should just say that it is incorrect. By the way, the function `round` with only one argument rounds to the nearest integer.

(a) $\frac{50}{21}$ type: int, value: 2
(b) $7\%4/1.0$ type: float, value 3.0
(c) '6'+'3'*5 type: string, value: '633333'
(d) $6 + '3'*5$ incorrectly formed
(e) 'computer' > 'computes' type: bool, value False
(f) (round(math.sqrt(10))>3) and (2**3 <10) type: bool, value: False

2. Consider the following function definition. (The function abs returns the absolute value of its argument.)

```python
def whoami(x,y,z):
    if (x==y) and (y==z):
        return False
    elif (abs(x)==abs(y)) or (abs(x)==abs(z)) or (abs(y)==abs(z)):
        return True
    else:
        return False
```

(a) What are the values of the following expressions?

whoami(3,0,-3), whoami(2,2,2), whoami(-1,3,2)

respectively: True, False, False

(b) Rewrite whoami so that your new function definition does exactly the same thing but does not use if, elif, or else. (It can be done in a single very long line, but it is smarter to begin by creating a few variables of boolean type.)

```python
def whoami(x,y,z):
    allsame=(x==y) and (y==z)
    twosame=(abs(x)==abs(y)) or (abs(y)==abs(z)) or (abs(x)==abs(z))
    return (not allsame) and twosame
```
3. Consider the following function definition:

```python
def whoami2(n):
    j=1
    while(j<=n):
        if j**2 == n:
            return True 
        j+=1
    return False
```

Note: The original version of this problem was missing the statement j+=1, which would have caused the function to loop infinitely on any input value other than 1.

(a) What are the values of the following expressions?

whoami2(1), whoami2(2*5), whoami2(5/2*8)

Respectively True, False, True

(b) Give a succinct description of what the function whoami2 does. (Don't answer the question by describing the code, with something like 'it assigns 1 to j, then it checks to see if j is less than or equal to the parameter n...' Instead tell me what problem the function is solving. What would be a good name for the function?)

The function returns True if the argument is the square of a nonzero integer, and False otherwise. A better name would be perfect_square.

4. Determine the output printed by the following code. (Recall that a print statement terminated by a comma does not advance to the next line, while the word 'print' all by itself on a line advances to the next line, but prints nothing.)

```python
j=1
r=1
while r<=5:
    c=1
    while c<= r:
        print j,
        j+=r
    c+=1
print r+=1
```
<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>9</td>
<td>12</td>
</tr>
<tr>
<td>15</td>
<td>19</td>
<td>23</td>
</tr>
<tr>
<td>31</td>
<td>36</td>
<td>41</td>
</tr>
</tbody>
</table>
5. In this problem you will have to correct the bugs in a program that contains a number of errors.

The gravitational force of the sun on an object of mass m kilograms at a distance d meters from the sun is:

\[ F = 1.33 \times 10^{12} \times \frac{m}{d^2} \]

where the mass is given in Newtons. (You don't have to know what any of the physics means.) A programmer writes a function that is supposed to return this force, given m and d as arguments.

```python
def gforce(m, d):
    print 1.33e12 * m/d*d
```

(a) The programmer tries to test the function, but when she tries to run it, she gets the message

There's an error in your program, invalid syntax

highlighting the first line of the code. What is the error, and how does she fix it?

A colon (:) at the end of the first line is missing.

(b) After fixing the syntax error, she then tries to call the function with some sample arguments, resulting in the following exchange with the Python Shell:

```python
>>> print gforce(1,2000)
1.33e+12
None
>>> print round(gforce(1,2000),2)
1.33e+12
```

Traceback (most recent call last):
  File "<pyshell#4>", line 1, in <module>
    print round(gforce(1,2000),2)
TypeError: a float is required
Something must be going wrong. What error in the function definition produced this behavior, and how can you fix it?

The function is called as though it is supposed to return a value. But it returns no value (hence the 'None') and produces an error when she attempts to pass the returned value to the function round. The solution is to replace the word 'print' in the function definition by 'return'.

(c) Finally, after the problem has been resolved, the programmer noticed that every input she tried with m=1 produced the same result 1.33e12, regardless of the value of d. What caused this error, and how did she fix it?

Because of the conventions on precedence of operations, \(1.33\text{e}12 \times \frac{m}{d^2}\) means \((1.33\text{e}12 \times m)/d\)*d, and not \(1.33\text{e}12m/(d^2)\), as required. The solution is to put parentheses around \(d^2\). You can also replace \(d\times d\) by \(d^2\), in which case parentheses are not necessary.