I commit to uphold the ideals of honor and integrity by refusing to betray the trust bestowed upon me as a member of the Georgia Tech community.

# CS 1301 Exam 1 Answers
## Fall 2009

<table>
<thead>
<tr>
<th>Problem</th>
<th>Earned Points</th>
<th>Possible Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Vocabulary</td>
<td>18</td>
<td>18</td>
</tr>
<tr>
<td>2. Python Expressions</td>
<td>26</td>
<td>26</td>
</tr>
<tr>
<td>3. Fill in the Blank</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>4. Multiple Choice</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>5. Reading Code</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>6. Combinations</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>7. MovieView</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>8. IsEven</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>9. Blastoff</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>10. Count J's</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td><strong>Total:</strong></td>
<td><strong>100</strong></td>
<td></td>
</tr>
</tbody>
</table>
1. Vocabulary (18 points)
For each of the following words, write a 1-2 sentence definition of the word as used in this class. Your definition should be concise and to the point, while proving you know what the term means.

a) evaluate - To calculate the value of an expression.
b) iteration - To repeatedly execute a block of code.
c) modulus - % operator, that calculates the remainder of an integer division.
d) proprioception - on a robot, internal sensing mechanisms. On a human, a sense of the relative positions of different parts of one's own body.
e) semantic error - An error that makes a program behave differently than intended by the programmer, but does not actually generate a syntax or runtime error.
f) type conversion - Deliberately changing the type of a value, using a function such as int(), str(), or float().

Grading: 3 points if the definition is very good, 2 points if they mostly got the concept, 1 point if they included a few of the right keywords, and zero if they get it completely wrong.

2. Python Expressions (26 points)
Act like the python interpreter and evaluate the following expressions. Write what value the expressions evaluate to as well as its type (integer, float, string, boolean, etc...).

<table>
<thead>
<tr>
<th>Expression</th>
<th>Evaluated Result (1 points)</th>
<th>Type of the Result (1 point)</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Hello” + “World” + “!”</td>
<td>“HelloWorld!”</td>
<td>String</td>
</tr>
<tr>
<td>5 + 6</td>
<td>11</td>
<td>Int</td>
</tr>
<tr>
<td>“cs1301” * 2</td>
<td>“cs1301cs1301”</td>
<td>String</td>
</tr>
<tr>
<td>int(5.9) / 3</td>
<td>1</td>
<td>Int</td>
</tr>
<tr>
<td>(6.0 – 1) ** 2 + 3</td>
<td>28.0</td>
<td>Float</td>
</tr>
<tr>
<td>“92” + str(34) + “Four”</td>
<td>“9234Four”</td>
<td>String</td>
</tr>
<tr>
<td>True and (3 != 2)</td>
<td>True</td>
<td>Boolean</td>
</tr>
<tr>
<td>range(3,7)</td>
<td>[3,4,5,6]</td>
<td>List</td>
</tr>
<tr>
<td>(7.0 + 6) / 2</td>
<td>6.5</td>
<td>Float</td>
</tr>
<tr>
<td>range(3,10,2)</td>
<td>[3,5,7,9]</td>
<td>List</td>
</tr>
<tr>
<td>5.0 &gt; 5.0</td>
<td>False</td>
<td>Boolean</td>
</tr>
<tr>
<td>“Pumpkin %.3f” % 3.1459</td>
<td>&quot;Pumpkin 3.146&quot;</td>
<td>String</td>
</tr>
<tr>
<td>7 + 3 / 2 &gt; 8</td>
<td>False</td>
<td>Boolean</td>
</tr>
<tr>
<td>(raw_input() &gt; 3) and False</td>
<td>False</td>
<td>Boolean</td>
</tr>
</tbody>
</table>
3. Fill in the Blank (11 points)

The name of my grading TA is: ___<Appropriate Name>____ and I am in section ___<SECTION>__.
For homeworks 2 & 3 my partner was: _____<NAME>_____.
In Python, a = is used for ___assignment___, while a == is used for ___equivalence checking___.
When a function calls itself, it is said to be ___Recursive / Recursion__________.
In python, the if keyword is used to make a ___conditional___ statement, while the for and while keywords are used to make ___loops (1/2 point: iteration)___.
The decimal (base 10) number { 43 }\textsubscript{10} is represented as ___ 101011___ in binary.
The binary number { 100101 }\textsubscript{2} is represented as ___37_______ in decimal.
The number { 28 }\textsubscript{10} is represented as _______1C_______ in hexadecimal.

4. Multiple Choice: Circle the correct choice. (6 points)

4a. Ada Lovelace is widely regarded as the first:
   A. Computer Scientist   **B. Programmer**   C. Compiler   D. Discrete Mathematician   E. None of these

4b. Douglas Engelbart demonstrated the world's first ________ in 1964 at Stanford.
   A. Transistorized Computer   **B. Mouse**   C. Solid State Memory   D. Tape Drive   E. Transistor

4c. Grace Hopper was:
   A. A Rear Admiral.   **B. Awarded the “man-of-the-year” award from DPMA in 1969.**
   C. Instrumental in the development of COBOL.   D. Credited for developing the first compiler.
   **E. All of the above.**

**Grading: 2 points for each correct answer.**
5. Reading Code (5 points)
Act like the python interpreter and run the following program. Write what would be printed out by the program in the box to the left.

```python
def fun1(x):
    print "Fun1 x: ", x
    return x * 2

print "Start"
y = fun1(10)
if (5 > y):
    print y
elif (15 > y):
    print y + 10
elif (25 > y):
    print y + 100
elif (35 > y):
    print y + 1000
else:
    print y + 10000
print "End"
```

Grading: 1 point for each line they get in the correct order, two points for getting "Fun1 x: 10"

6. Writing Code - Combinations (5 points)
Write a function named `combine` that accepts two strings as parameters. The function should concatenate the strings (putting the second parameter after the first parameter), and return the new combined string.

```python
def combine(string1, string2):
    return (string1 + string2)
```

Grading: +1 point for getting the header correct.  
+2 point for adding the two strings,  
+2 point for returning the correct result
7. Correcting Code: MovieView (6 points)

Look over the following lines of code, and for each line determine if the code will run (syntax is correct). If the line of code is correct, just tell us "It works!", if it will not, please explain why/correct the error.

<table>
<thead>
<tr>
<th>Line</th>
<th>Code</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>1:</td>
<td><code>define movieView(age):</code></td>
<td>def instead of define</td>
</tr>
<tr>
<td>2:</td>
<td><code>    name = raw_input(&quot;What is your name? &quot;)</code></td>
<td>It works!</td>
</tr>
<tr>
<td>3:</td>
<td><code>    if age &lt; 17</code></td>
<td>Missing colon (:) after the 17</td>
</tr>
<tr>
<td>4:</td>
<td><code>        print &quot;Sorry %d, you can't see the movie alone.&quot; % name</code></td>
<td>%d needs to be a %s to accept a string (name)</td>
</tr>
<tr>
<td>5:</td>
<td><code>    else age &gt;= 17:</code></td>
<td>No boolean condition after the else, remove (age &gt;= 17)</td>
</tr>
<tr>
<td>6:</td>
<td><code>        print &quot;Enjoy the film, %s!&quot; % name</code></td>
<td>It works!</td>
</tr>
</tbody>
</table>

8. Write Code - IsEven (5 points)

Write a function called `IsEven` that accepts a single parameter `aNum`. If `aNum` is even (evenly divisible by 2) the function must return True, otherwise, it must return False.

```python
def IsEven( aNum):
    if (aNum % 2 == 0):
        return True
    else:
        return False
```

Grading:
+1 point for getting the header correct
+2 points for determining if the value is even/odd
+2 points for returning the correct boolean
9. Write Code - Blastoff (8 points)
Write a function called blastoff that will accept a single parameter called N. Your function should start counting down from N, printing each number one per line until N reaches zero. When N reaches zero, instead of printing "0", the function should print "Blastoff!". For example, if you called blastoff and gave it an N of 5, this would be the result:
```>>> blastoff(5)
5
4
3
2
1
Blastoff!
``` 
```def blastoff( N ):
    while (N > 0):
        print N
        N = N - 1
    print "Blastoff!"
``` 
Grading:
+2 point for getting the header correct
+4 points for printing each number from N to 1 (+2 if they miss one number at either end)
+2 point for printing "Blastoff" at the end

10. Write Code - Count Js (10 points)
Write a function named countJs that accepts a single string parameter called aString. The countJs function should keep a count of how many times the capital letter "J" appears in the string, and return the final count.
```def countJs( aString ):
    counter = 0
    for letter in aString:
        if( letter == "J"):
            counter = counter + 1
    return( counter)
``` 
Grading:
+2 for getting the header correct
+2 for having a counter that starts at zero
+2 for comparing each element to a "J"
+2 for incrementing the counter if it was a "J"
+2 for returning a result