• **INSTRUCTIONS:**
  
  – Keep your eyes on your own paper and do your best to prevent anyone else from seeing your work.
  – Do NOT communicate with anyone other than the professor/proctor for ANY reason in ANY language in ANY manner.
  – This exam is closed notes, closed books, and no calculator.
  – Turn all mobile devices off and put them away now. You cannot have them on your desk.
  – Write neatly and clearly indicate your answers. What I cannot read, I will assume to be incorrect.
  – Stop writing when told to do so at the end of the exam. I will take 5 points off your exam if I have to tell you multiple times.
  – Academic misconduct will not be tolerated. Suspected academic misconduct will be immediately referred to the Emory Honor Council. Penalties for misconduct will be a zero on this exam, an F grade in the course, and/or other disciplinary action that may be applied by the Emory Honor Council.

• **TIME:** This exam has 6 questions on 7 pages including the title page. Please check to make sure all pages are included. You will have 50 minutes to complete this exam.

---

*I commit to uphold the ideals of honor and integrity by refusing to betray the trust bestowed upon me as a member of the Emory community. I have also read and understand the requirements and policies outlined above.*

**Signature:**

---

<table>
<thead>
<tr>
<th>Question</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Points:</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>6</td>
<td>12</td>
<td>50</td>
</tr>
<tr>
<td>Score:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
1. **Short Answer**

(a) (1 point) Give an example of an active sensor on your robot.

**Solution:** IR, obstacle

(b) (1 point) Give an example of a passive sensor on your robot.

**Solution:** light, brightness, battery

(c) (1 point) How is a cyborg different than a robot?

**Solution:** Cyborg has a biological component instead of being purely mechanical.

(d) (1 point) In python, a single equals symbol is used to indicate ____________ while a double equals sign is used to indicate ____________

**Solution:** assignment, equality

(e) (1 point) A ____________ causes execution to leave the current function and resume at the point in the code immediately after where the function was called. It also allows a function to specify a value to be passed back to the code that called the function.

**Solution:** return statement

(f) (1 point) You use ____________ in your python programs, denoted by the # symbol, to explain in natural language how your program works.

**Solution:** comment

(g) (2 points) Robots use effectors and actuators to manipulate the environment. What is the difference between these components? Give an example of each. (Does not have to be Scribbler specific.)

**Solution:** Effectors: any device that effects the environment
Ex: Arms, legs, wheels, fingers, hooks, wings, etc.
Actuator: actual mechanism that enables effector to execute actions
Ex: electric motors (current), hydraulic cylinders (fluid pressure), pneumatic cylinders (air pressure)
2. Code tracing.
   For each of the following snippets of code, indicate what would be printed.
   
   (a) (4 points) `def print1():`
       `print 1`
       `return 1`
   
       `def print2():`
       `return 2`
       `print 2`
   
       `v = print1()`  
       `w = print2()`  
       `print v`  
       `print w`  
   
       **Solution:** 1 1 2 (each on separate line)
   
   (b) (4 points) `mylist = ["apple", "banana", "grape", "cherry", "orange", "peach"]`
   `for i in range(1, 5, 2):`
       `print mylist[i], mylist[i/2]`

   **Solution:** banana apple  
   cherry banana
3. (8 points) List manipulation.

The statements below are executed in order in the python interpreter. For each print statement, give the output. If the print statement would cause an error, you may just write “error”.

```python
defun mylist = ['cs190', "40", -30.6, "V", 50]
defun print mylist[3]
    V
print mylist[5]
    error (index out of range)
    error (can't concat float and str)
print mylist[2:4]
    [-30.6, "V"]
mylist[1] = int(mylist[1])
mylist[0] = mylist[1] + 20
mylist[3] = 200
print mylist
    [60, 40, -30.6, 200, 50]
x = mylist[2]
mylist[3] = x
print mylist
    [60, 40, 200, -30.6, 50]
mylist.append(2)
print mylist
    [60, 40, 200, -30.6, 50, 2]
print len(mylist)
    6
```
4. (8 points) Evaluate each expression. Then give the result of the evaluation and the data type of the result. If the expression cannot be evaluated due to an error, you may simply write “error” for the value. The first row has been done for you.

For the following expressions, assume the following line of code has been executed:

```python
word = "CS 190"
```

<table>
<thead>
<tr>
<th>Expression</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>True or False</td>
<td>True</td>
</tr>
<tr>
<td>not(True or False) and True</td>
<td>False</td>
</tr>
<tr>
<td>3 != 2*3-3</td>
<td>False</td>
</tr>
<tr>
<td>6 &gt;= 2*3.0 and 4 == 2**2</td>
<td>True</td>
</tr>
<tr>
<td>(4+2 &gt; 3 and True) != True</td>
<td>False</td>
</tr>
<tr>
<td>4 &lt; 6 &gt; 10</td>
<td>False</td>
</tr>
<tr>
<td>word[0]</td>
<td>“C”</td>
</tr>
<tr>
<td>word[3:5]</td>
<td>“19”</td>
</tr>
<tr>
<td>’9’ in word</td>
<td>True</td>
</tr>
</tbody>
</table>
5. (6 points) A pentagonal number is defined as \( \frac{n(3n-1)}{2} \) for \( n = 1, 2, \ldots \) and so on. So the first few numbers in a pentagonal series are 1, 5, 12, 22 ..... Write a function named \texttt{pentagonal} which takes one input parameter and returns the \( n \)th pentagonal number. Examples of function calls:

\[
\begin{align*}
\text{pentagonal}(2) & \rightarrow 5 \\
\text{pentagonal}(4) & \rightarrow 22
\end{align*}
\]

\textbf{Solution:}

\[
\begin{align*}
def \text{pentagonal}(n):
    p &= (n * (3 * n - 1))/2 \\
    \text{return } p
\end{align*}
\]

Scoring: +2 function definition  
+2 math correct  
+2 return statement correct
6. (12 points) Write a function named `averageElements` that takes two parameters: a list and an integer. Function should compute the average of the first $n$ elements of the list. You are guaranteed that the list will have at least $n$ elements in it and all data in the list will be numerical. Example function calls:

```python
averageEls([3, 3, 6, 4], 3) -> 4.0
averageEls([70, 71, 90, 100], 2) -> 70.5
```

Solution:

```python
def averageElements(mylist, n):
    sum = 0.0
    for i in range(n):
        sum = sum + mylist[i]
    avg = sum / n
    return avg
```

Scoring:
+3 function definition
+3 correct use of loop (n iterations of loop)
+2 sums elements correctly
+2 computes average correctly
+1 return statement used correctly
+1 int/floating point issues accounted for