# Assignment Answers - Assertions and Logging

## 1. Create an assert statement that throws an AssertionError if the variable spam is a negative integer.

```python  
assert spam >= 0, 'spam should not be negative'  
```

## 2. Write an assert statement that triggers an AssertionError if the variables eggs and bacon contain strings that are the same as each other, even if their cases are different.

```python  
assert eggs.lower() != bacon.lower(), 'eggs and bacon should not be the same'  
```

## 3. Create an assert statement that throws an AssertionError every time.

```python  
assert False, 'This assertion always fails'  
```

## 4. What are the two lines that must be present in your software in order to call logging.debug()?

```python  
import logging  
logging.basicConfig(level=logging.DEBUG)  
```

## 5. What are the two lines that your program must have in order to have logging.debug() send a logging message to a file named programLog.txt?

```python  
import logging  
logging.basicConfig(filename='programLog.txt', level=logging.DEBUG)  
```

## 6. What are the five levels of logging?

1. DEBUG  
2. INFO  
3. WARNING  
4. ERROR  
5. CRITICAL

## 7. What line of code would you add to your software to disable all logging messages?

```python  
logging.disable(logging.CRITICAL)  
```

## 8. Why is using logging messages better than using print() to display the same message?

Logging allows different severity levels, writes to files, and can be disabled easily, while print() is always visible and not as flexible.

## 9. What are the differences between the Step Over, Step In, and Step Out buttons in the debugger?

- \*\*Step Over\*\*: Executes the function call without stepping into it.  
- \*\*Step In\*\*: Steps into the function to debug inside it.  
- \*\*Step Out\*\*: Exits the current function and returns to the caller.

## 10. After you click Continue, when will the debugger stop?

The debugger stops when it hits the next breakpoint or the program ends.

## 11. What is the concept of a breakpoint?

A breakpoint is a marker in the code where the debugger pauses execution, allowing inspection of variables and program state.