**Assignment 11**

**1. Create an assert statement that throws an Assertion Error if the variable spam is a negative**

**integer.**

**Ans-1** In Python programming, assert statements declare whether a condition is true prior to run the code, and the control simply moves to the next line if it is true.

Variable spam > 0

**2. Write an assert statement that triggers an Assertion Error if the variables eggs and bacon contain**

**strings that are the same as each other, even if their cases are different (that is, hello and hello are**

**considered the same, and goodbye and GOODbye are also considered the same).**

**Ans-2** Variable eggs.lower() != bacon.lower()- In this case The eggs and bacon variables are the same

Variable eggs.upper() != bacon.upper()-- In this case The eggs and bacon variables are the same

**3. Create an assert statement that throws an Assertion Error every time.**

**Ans-3** assert False- This assertion statement always assert error

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

**Ans-4** To be able to call logging.debug(), you must have these two lines at the start of your program:

import logging

logging.basicConfig(level=logging.DEBUG, format=' %(asctime)s - %(levelname)s - %(message)s')

**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?**

**Ans-5 To** be able to send logging messages to a file named programLog.txt with logging.debug(), you must have these two lines at the start of your program:

import logging

logging.basicConfig(filename='programLog.txt', level=logging.DEBUG, format=' %(asctime)s - %(levelname)s - %(message)s')

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

**Ans-6 The five levels of logging are**

* ERROR
* DEBUG
* WARNING
* CRITICAL
* INFO

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

**Ans-7** To disable all logging messages we use

logging.disable(logging.CRITICAL)

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

**Ans-8** A logging message can be disabled without removing the logging function call, a logging message can be selectively disabled, a logging message can be created. Logging messages provides a timestamp.

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

**Ans-9** The Step button will move the debugger into a function call. The Over button will quickly execute the function call without stepping into it. The Out button will quickly execute the rest of the code until it steps out of the function it currently is in.

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

**Ans-10** After you click Go, the debugger will stop when it has reached the end of the program or a line with a breakpoint.

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

**Ans-11** A breakpoint is a setting on a line of code that causes the debugger to pause when the program execution reaches the line.