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

Sol: assert spam>=0

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 (that is, 'hello' and 'hello' are considered the same, and 'goodbye' and 'GOODbye' are also considered the same).

Sol: assert eggs.lower()!=bacon.lower()

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

Sol: assert False

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

Sol: 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?

Sol: import logging

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

6. What are the five levels of logging?

Sol: 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?

Sol: logging.disable(logging.CRITICAL)

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

Sol: because logging provides control over message severity and destination and it will be easy for the debugging and troubleshooting when ever we want.

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

Sol: step over: the step over button executes the current line of code and then stops at the next line of code. If the current line contains a function call, the function is executed in it’s entirety, but the debugger does not stop on each individual line of the function

Step in: If the current line contains a function call,the debugger steps into the function and stops on the first line of the function, allowing you to step through the function line-by-line.

Step out: The step out button executes the remaining lines of the current function and stops at the line of code that called the function.

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

Sol: After we click Continue the debugger stops when breakpoint reached, exception raised, end of program reached.

11. What is the concept of a breakpoint?

Sol: A breakpoint is a point in your code where the debugger will temporarily pause the execution of your program, allowing you to inspect the program state, step through the code line-by-line, and diagnose problems with the program’s behavior.