1. assert(spam<0, ‘ The spam, variable is negative’)
2. assert(eggs.lower() != bacon.lower(), 'The eggs and bacon variables are the same!') or assert(eggs.upper() != bacon.upper(), 'The eggs and bacon variables are the same!')
3. assert(False, 'This assertion always triggers.')
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. import logging logging.basicConfig(filename='programLog.txt', level=logging.DEBUG, format=' %(asctime)s - %(levelname)s - %(message)s')
6. DEBUG, INFO, WARNING, ERROR, and CRITICAL
7. logging.disable(logging.CRITICAL)
8. You can disable logging messages without removing the logging function calls. You can selectively disable lower-level logging messages. You can create logging messages. Logging messages provides a timestamp.
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. Click Continue, the debugger will stop when it has reached the end of the program or a line with a breakpoint.
11. Set a breakpoint in IDLE, right-click the line and select Set Breakpoint from the context menu.