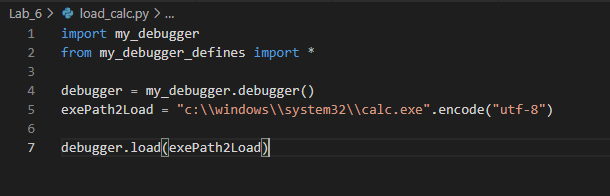
NOTE – since it did not specify what you wanted for the submissions for the problems, I added screenshots of each program you asked us to create, I also will add the altered files of my\_debugger and my\_debugger\_defines

Problem 1:



Question 1:

1. a. It loads the kernel32 dll

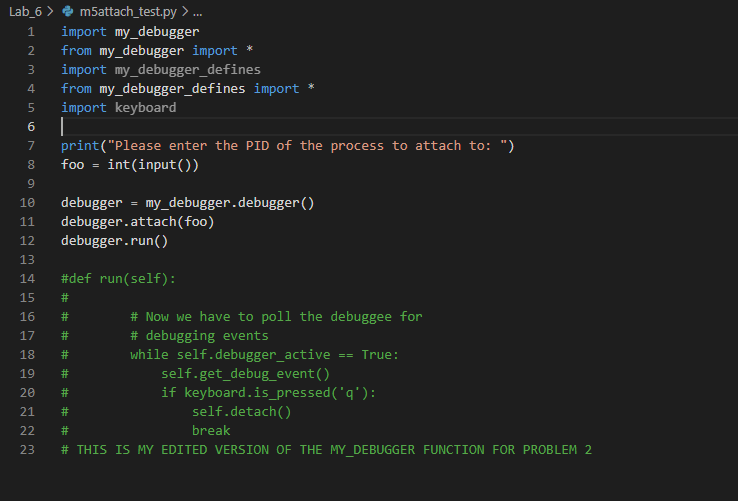
b. retrieves system information regarding the passed pointer inside byref

1. SYSTEM\_INFO is a class defined in my\_debugger\_defines.py
2. Pointers refer to addresses of data in memory handles refer to an object or program
3. a. Creates a new process and its primary thread.

b. Opens an existing local process object.

1. In my\_debugger.load you need to set creation\_flags = CREATE\_NEW\_CONSOLE to see the GUI
2. Yes you can, when I tested it I got the Process ID 7592

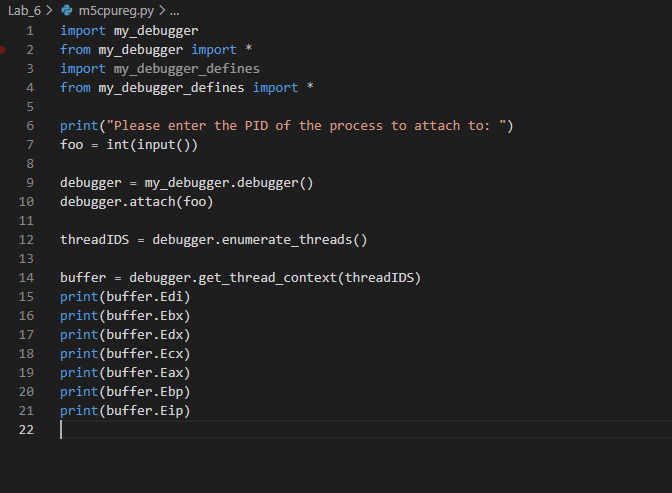
Problem 2:



Question 2:

1. Event Code: 3, which means CREATE\_PROCESS\_DEBUG\_EVENT, which is what happens when a process is created
2. Yes, you can interact with the process (which incidentally causes lots of event codes to trigger)
3. A debug event has occurred when the debugger prints out an event code to the terminal

Problem 3:



Question 3:

1. A thread is a single sequential flow of control within a program.
2. A thread context is the information the thread needs to resume execution, including the thread's set of CPU registers and stack.
3. It will not be the same, because multithreaded processers don’t share registers
4. Same as the answer above.

NOTE – I did not provide screenshots for the registers as I could not get them to output anything but zero. I changed the values in my\_debugger\_defines from DWORDS to QWORDS but tried a bunch of different things but nothing worked.