Author: Dominic Fantauzzo

ONID ID number: 934-556-579

Date: 10/21/2023

OSU email address: fantauzd@oregonstate.edu

Course number/section: CS 271, 400

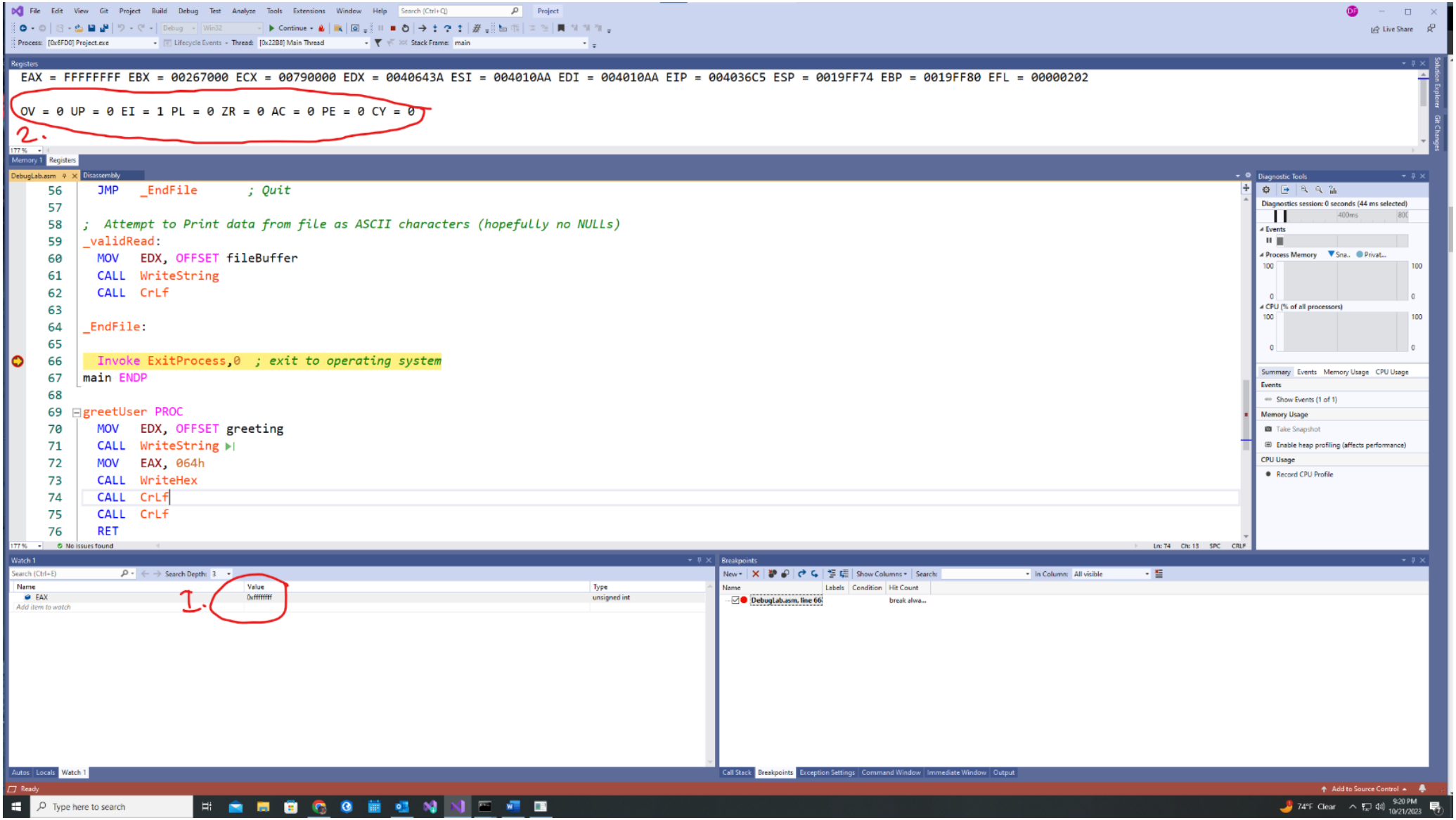
Project Number: 2 – Debugging Lab

Part 1:

Question 1: The current value (in Hex) is `0xffffffff`

Question 2: Carry: clear, Overflow: clear, Zero: clear

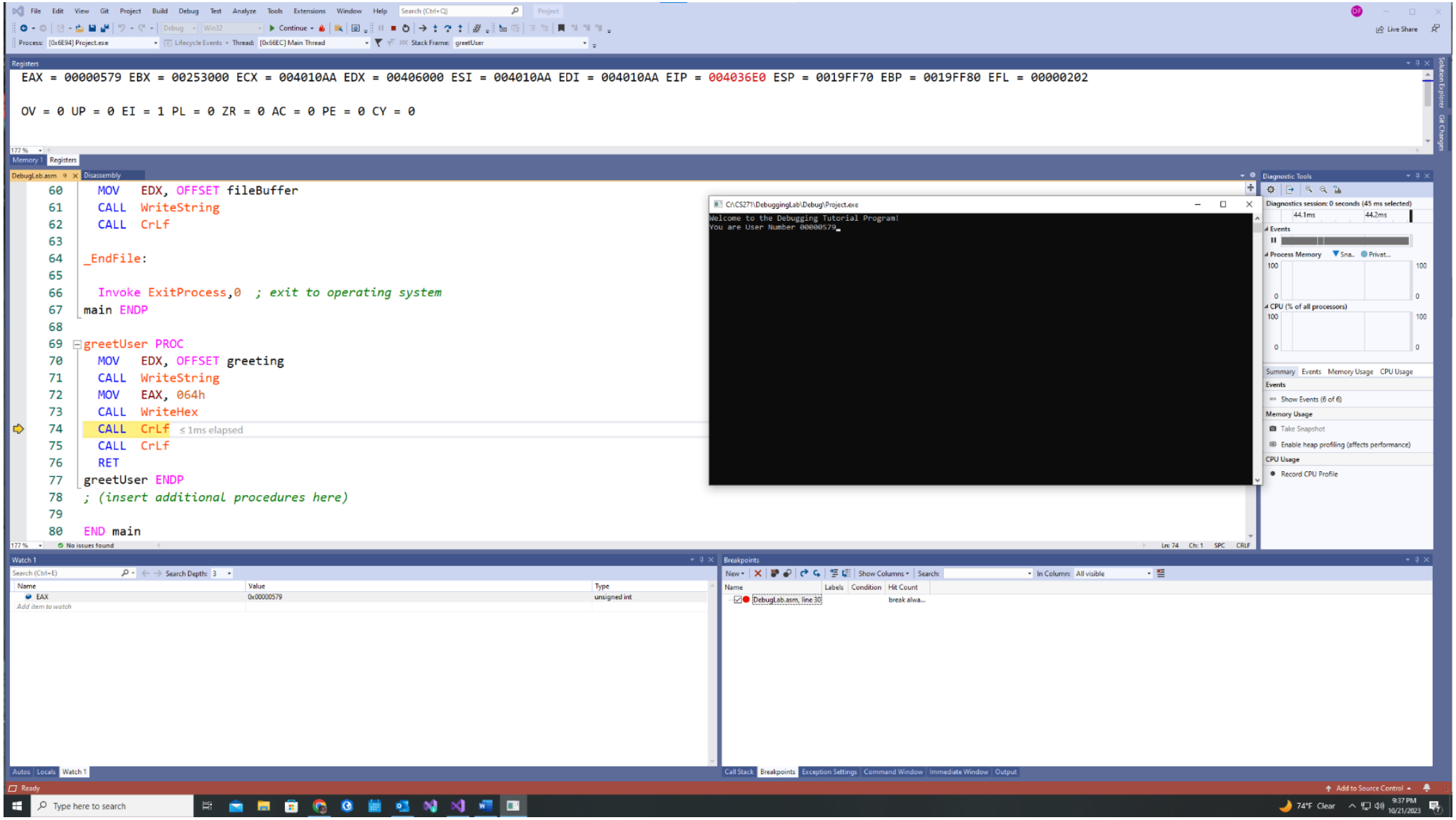
Screenshot 1: Part 1, Questions 1-2



Part 2:

Question 1: *shown below*

Screenshot 2: Part 2, Question 1



Part 3:

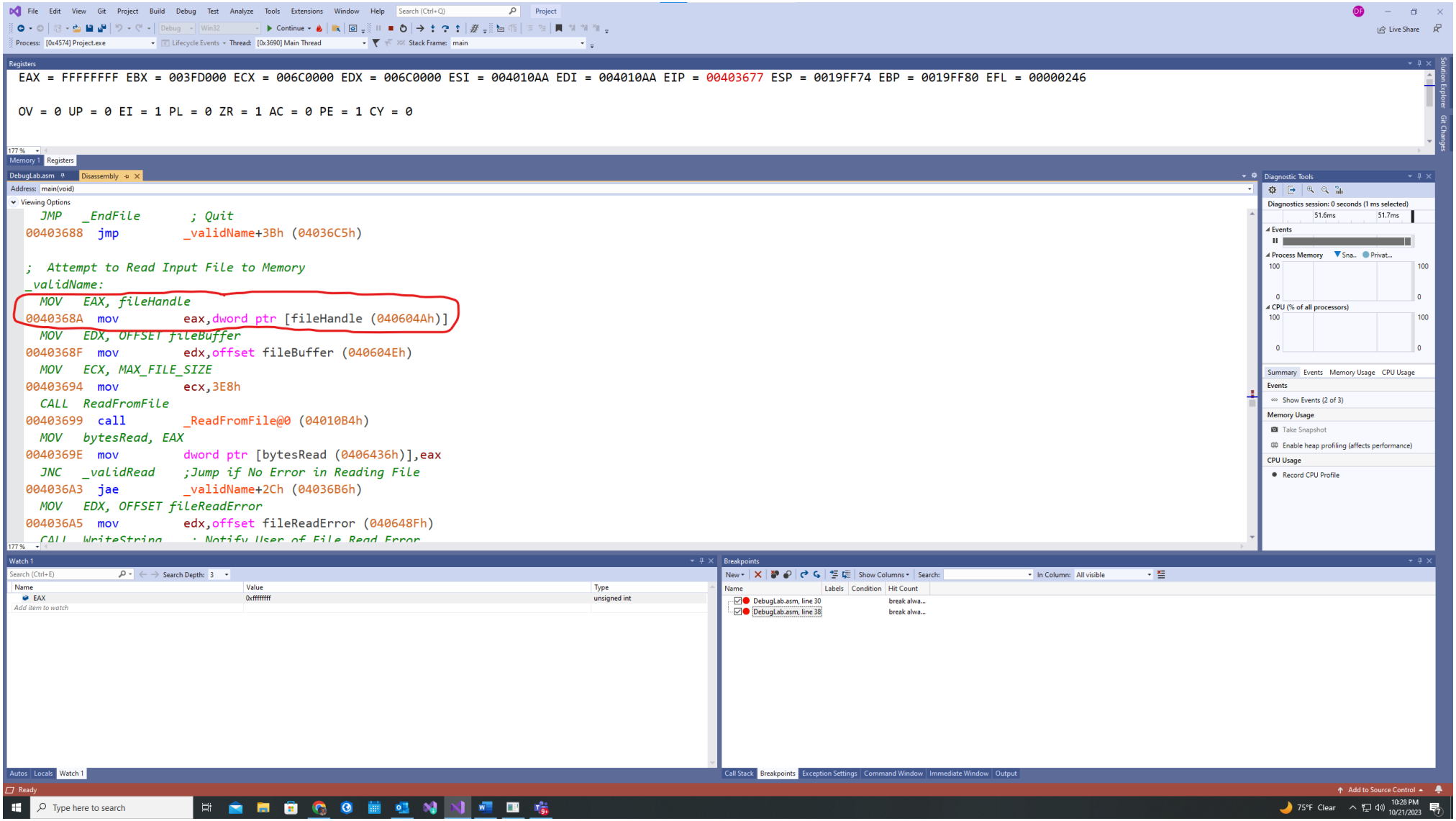
Question 1: The instruction at the _validName code label is mov. It has a destination operator, eax, and a source operator fileHandle.

Question 2: The memory address is 0040368A

Question 3:

The value in EIP represents the memory address of the next instruction that will be executed. We can track this value by using the registers window when debugging. The leftmost value on any given line represents the memory address for that instruction (when in disassembly, as shown below). Hence, the leftmost value on any given line is fed into the EIP register when that line is the next to be executed. Therefore, the relationship between the value in EIP and the leftmost value on any given line is significant because we, as the programmer, can use the debugger to compare the leftmost value to EIP and determine which line the code will execute next. This gives us the ability to see if our flow of control is executing as expected, especially when debugging branching and looping structures. If we run into bugs, we can utilize this relationship to step through code and find where EIP does not behave as expected. Pinpointing where EIP does not match our expected leftmost value can allow us to fix issues more quickly.

Screenshot 3: Part 3, Question 1-2

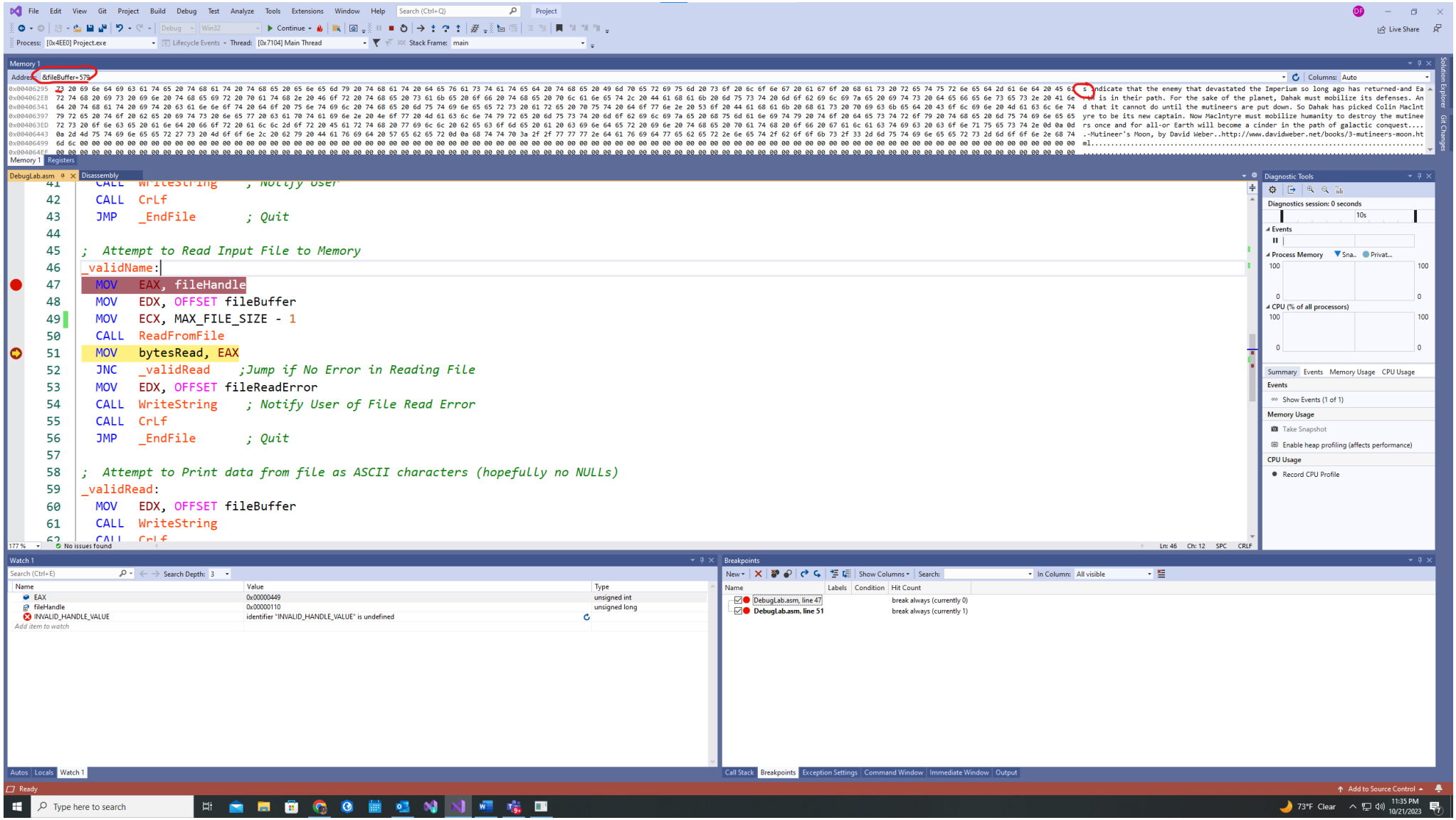


Part 4:

Question 1: My last three OSU ID digits are 579. The 580th byte (index 579) of TestText.txt, as an ASCII character, is s. (0x73)

Question 2: *see below*

Screenshot 4: Part 4, Question 1



Part 5:

Question 1: As the OpenInputFile procedure returns the file handle in the EAX register, which is then moved to the fileHandle indentifier, the file handle of TestText,txt in my execution of the lab program must be `0x00000108`.

Screenshot 5: Part 5, Question 1