

## Project 2 – Using the Debugger (Lab)

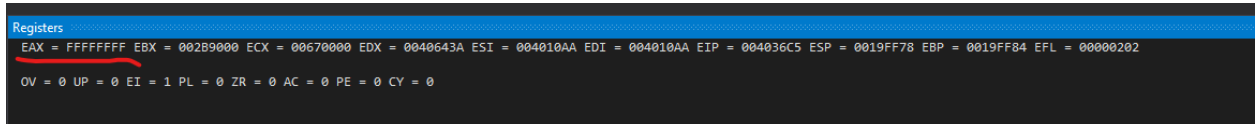
Name: Casey Lowe

ONID #: 934494518

Date: 7/12/2023

### Part 1 Questions:

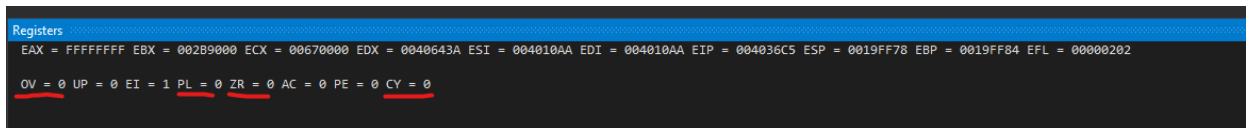
1. What is the current value (in Hex) of the EAX Register?



```
Registers
EAX = FFFFFFFF EBX = 002B9000 ECX = 00670000 EDX = 0040643A ESI = 004010AA EDI = 004010AA EIP = 004036C5 ESP = 0019FF78 EBP = 0019FF84 EFL = 00000202
OV = 0 UP = 0 EI = 1 PL = 0 ZR = 0 AC = 0 PE = 0 CY = 0
```

- a.
- b. EAX = FFFFFFFF

2. What is the current state (Set/Clear) of the following flags: Carry, Overflow, Zero, Sign?

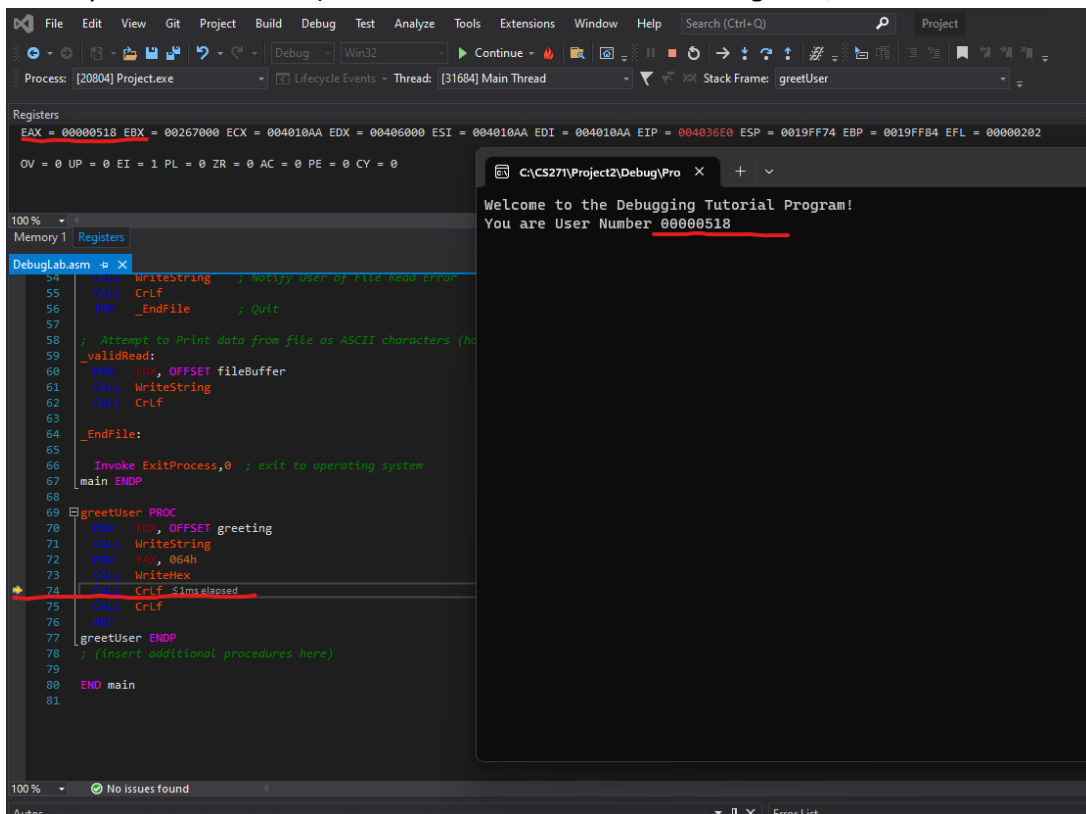


```
Registers
EAX = FFFFFFFF EBX = 002B9000 ECX = 00670000 EDX = 0040643A ESI = 004010AA EDI = 004010AA EIP = 004036C5 ESP = 0019FF78 EBP = 0019FF84 EFL = 00000202
OV = 0 UP = 0 EI = 1 PL = 0 ZR = 0 AC = 0 PE = 0 CY = 0
```

- a.
- b. Carry = 0
- c. Overflow = 0
- d. Zero = 0
- e. Sign = 0

### Part 2 Questions

1. (Screenshot of your User Number (number in Terminal Window and Registers/Editor window in Visual Studio).

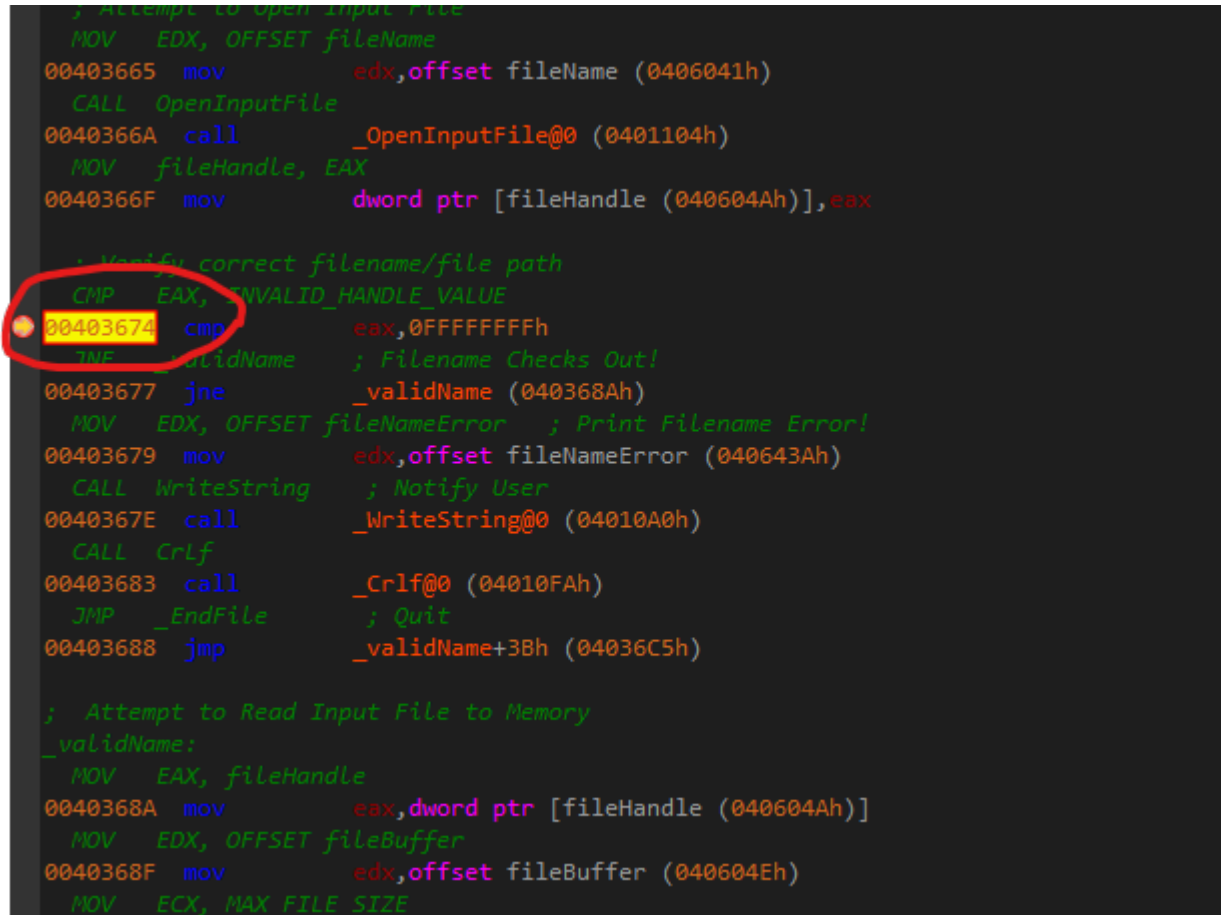


```
File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Help Search (Ctrl+Q) Project
Process: [20804] Project.exe Thread: [31684] Main Thread Stack Frame: greetUser
Registers
EAX = 00000518 EBX = 00267000 ECX = 004010AA EDX = 00406000 ESI = 004010AA EDI = 004010AA EIP = 004036E0 ESP = 0019FF74 EBP = 0019FF84 EFL = 00000202
OV = 0 UP = 0 EI = 1 PL = 0 ZR = 0 AC = 0 PE = 0 CY = 0
Memory | Registers
DebugLab.asm
54 CALL WriteString ; Write user ID file read from
55 CALL Crlf
56 JMP _EndFile ; Quit
57
58 ; Attempt to Print data from file as ASCII characters (hex)
59 _validRead:
60 MOV EDI, OFFSET fileBuffer
61 CALL WriteString
62 CALL Crlf
63
64 _EndFile:
65
66 Invoke ExitProcess,0 ; exit to operating system
67 main ENDP
68
69 greetUser PROC
70 MOV EDI, OFFSET greeting
71 CALL WriteString
72 MOV EAX, 064h
73 CALL WriteHex
74 CALL Crlf ; Simulated
75 CALL Crlf
76 RET
77 greetUser ENDP
78 ; (insert additional procedures here)
79
80 END main
81
100% No issues found
Autos Error List
```

- a.

### Part 3 Questions

1. What instruction (with operands, if any) is located at the `_validName` code label? (Hint: The first instruction after a code label is "at" that label.)
  - a. `CMP EAX,0FFFFFFFh`
2. What is the memory address (in the code segment) of the instruction from Question 1 above? (Hint: It will be a hex value to the left of this instruction.) Please circle this line on the screenshot attached to this part in your report, including the instruction and operands from Question 1.



```
; Attempt to Open Input File
MOV EDX, OFFSET fileName
00403665 mov     edx,offset fileName (0406041h)
CALL OpenInputFile
0040366A call     _OpenInputFile@0 (0401104h)
MOV fileHandle, EAX
0040366F mov     dword ptr [fileHandle (040604Ah)],eax

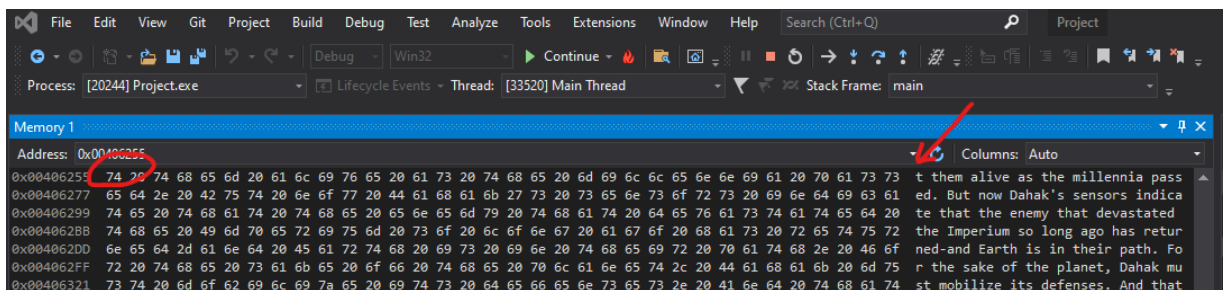
; Verify correct filename/file path
CMP EAX, INVALID_HANDLE_VALUE
00403674 cmp     eax,0FFFFFFFh
JNE _validName ; Filename Checks Out!
00403677 jne     _validName (040368Ah)
MOV EDX, OFFSET fileNameError ; Print Filename Error!
00403679 mov     edx,offset fileNameError (040643Ah)
CALL WriteString ; Notify User
0040367E call     _WriteString@0 (04010A0h)
CALL Crlf
00403683 call     _Crlf@0 (04010FAh)
JMP _EndFile ; Quit
00403688 jmp     _validName+3Bh (04036C5h)

; Attempt to Read Input File to Memory
_validName:
MOV EAX, fileHandle
0040368A mov     eax,dword ptr [fileHandle (040604Ah)]
MOV EDX, OFFSET fileBuffer
0040368F mov     edx,offset fileBuffer (040604Eh)
MOV ECX, MAX_FILE_SIZE
```

- a.
  - b. 00403674
3. What is the significance of the relationship between the value in EIP (in the Registers window) and the leftmost value on any given line?
    - a. The value in EIP and the left most value on any given line should be equivalent. If they are not then there is something wrong, and the programmer needs to go back and debug the issue.

## Part 4 Questions

- Let  $n$  = (the last three digits of your OSU ID number). What is the  $(n+1)$ st byte (index  $n$ ) of TestText.txt, interpreted as an ASCII character?



- $N = 518 \rightarrow \&\text{fileBuffer} + 518 + 1$
- Remember to take a screenshot of the Memory window used to collect this information for inclusion in the lab report. Please circle the value in the address bar and the character in the ASCII portion of the Memory debug window.

## Part 5 Questions

- A file handle is a reference number supplied by the operating system and used by a running program to uniquely identify a file. The lab uses the fileHandle variable to store this value. What is the file handle of TestText.txt in your execution of the lab program? (This builds on the same watch you used in Step 6 above but is a new question. There are multiple 'watches' which can get you this value, depending on where you have paused execution.)

Please be aware that the handle for a particular file may be different each time you run your program (the operating system state may change between runs), so remember to include a screenshot that shows where you obtained this information, and please circle the value of the file handle on this screenshot.

