

NAME: OLORUNFEMI-OJO DANIEL TOMIWA

MATRIC: 190805503

DEPARTMENT: COMPUTER SCIENCE

COURSE: CSC 315

SIGN:

TITLE Detecting the highest of 2 numbers
; Author: Olorunfemi-Ojo Daniel Tomiwa 190805503

.386

.MODEL FLAT, STDCALL

.STACK

ExitProcess Proto, dwExitCode:DWORD

.DATA

val1 DWORD 74

val2 DWORD 10

.CODE

main PROC

MOV EAX, val1

MOV EBX, val2

CMP EAX, EBX

JA firstnum

MOV ECX, EBX ; copy EBX to ECX if higher

JMP finish

firstnum:

MOV ECX, EAX ; copy EAX to ECX if higher

finish:

INVOKE ExitProcess, 0

main ENDP

END main

EXAMPLE 1: 7 and 3

```
1 TITLE Detecting the highest of 2 numbers
2 ; Author: Olurunfemi-Ojo Daniel Tamiwa 190805503
3
4 .386
5 .MODEL FLAT, STDCALL
6 .STACK
7 ExitProcess PROTO, dwExitCode:DWORD
8
9 .DATA
10 val1 DWORD 7
11 val2 DWORD 3
12 .CODE
13 main PROC
14 MOV EAX, val1
15 MOV EBX, val2
16 CMP EAX, EBX
17 JA firstnum
18 MOV ECX, EBX ; copy EBX to ECX if higher
19 JMP finish
20 firstnum:
21 MOV ECX, EAX ; copy EAX to ECX if higher
22 finish:
23 INVOKE ExitProcess, 0
24
```

Registers:

EAX = 00000007 EBX = 00000003 ECX = 00461005
EDX = 00461005 ESI = 00461005 EDI = 00461005
EIP = 00461018 ESP = 012FFAA0 EBP = 012FFAAC
EFL = 00000246

OV = 0 UP = 0 EI = 1 PL = 0 ZR = 1 AC = 0 PE = 1
CY = 0

Locals:

Name	Value	Type
EAX	7	unsigned int
EBX	3	unsigned int
ECX	4591621	unsigned int

```
1 TITLE Detecting the highest of 2 numbers
2 ; Author: Olurunfemi-Ojo Daniel Tamiwa 190805503
3
4 .386
5 .MODEL FLAT, STDCALL
6 .STACK
7 ExitProcess PROTO, dwExitCode:DWORD
8
9 .DATA
10 val1 DWORD 7
11 val2 DWORD 3
12 .CODE
13 main PROC
14 MOV EAX, val1
15 MOV EBX, val2
16 CMP EAX, EBX
17 JA firstnum
18 MOV ECX, EBX ; copy EBX to ECX if higher
19 JMP finish
20 firstnum:
21 MOV ECX, EAX ; copy EAX to ECX if higher
22 finish:
23 INVOKE ExitProcess, 0
24
```

Registers:

EAX = 00000007 EBX = 00000003 ECX = 00461005
EDX = 00461005 ESI = 00461005 EDI = 00461005
EIP = 0046101D ESP = 012FFAA0 EBP = 012FFAAC
EFL = 00000202

OV = 0 UP = 0 EI = 1 PL = 0 ZR = 0 AC = 0 PE = 0
CY = 0

Locals:

Name	Value	Type
EAX	7	unsigned int
EBX	3	unsigned int
ECX	4591621	unsigned int

```
1 TITLE Detecting the highest of 2 numbers
2 ; Author: Olurunfemi-Ojo Daniel Tamiwa 190805503
3
4 .386
5 .MODEL FLAT, STDCALL
6 .STACK
7 ExitProcess PROTO, dwExitCode:DWORD
8
9 .DATA
10 val1 DWORD 7
11 val2 DWORD 3
12 .CODE
13 main PROC
14 MOV EAX, val1
15 MOV EBX, val2
16 CMP EAX, EBX
17 JA firstnum
18 MOV ECX, EBX ; copy EBX to ECX if higher
19 JMP finish
20 firstnum:
21 MOV ECX, EAX ; copy EAX to ECX if higher
22 finish:
23 INVOKE ExitProcess, 0
24
```

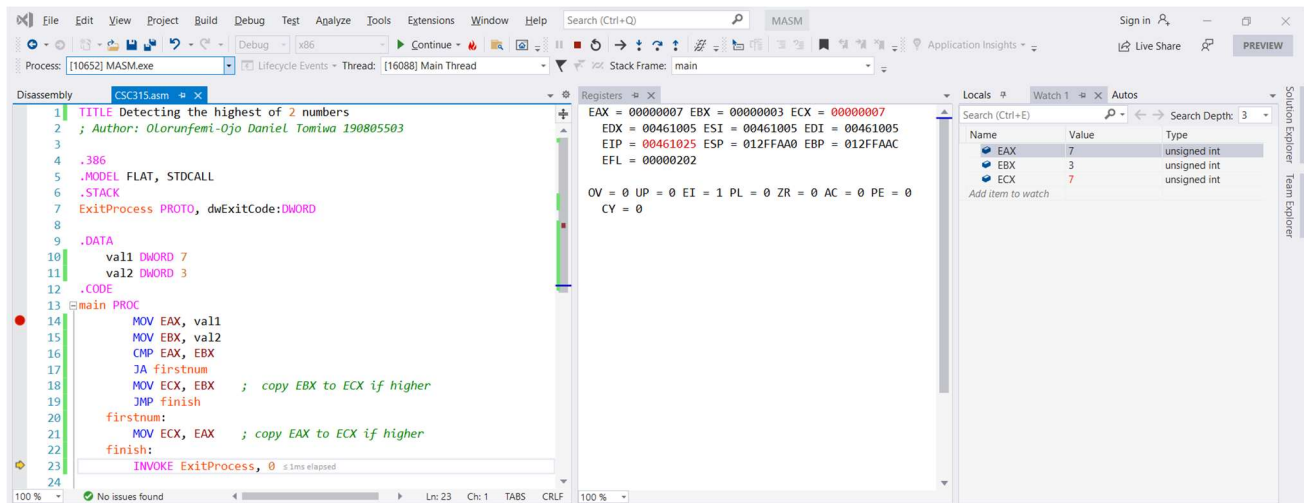
Registers:

EAX = 00000007 EBX = 00000003 ECX = 00461005
EDX = 00461005 ESI = 00461005 EDI = 00461005
EIP = 00461023 ESP = 012FFAA0 EBP = 012FFAAC
EFL = 00000202

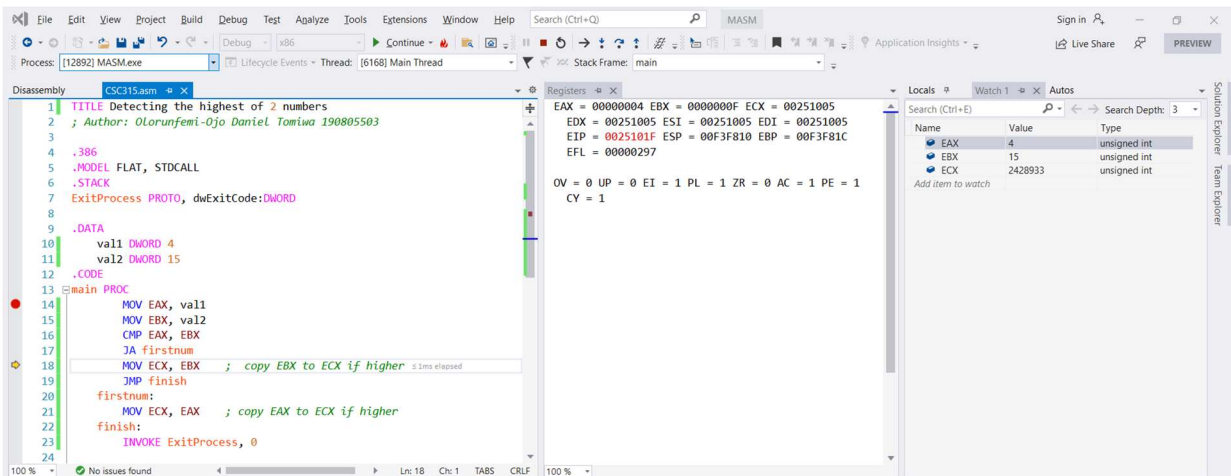
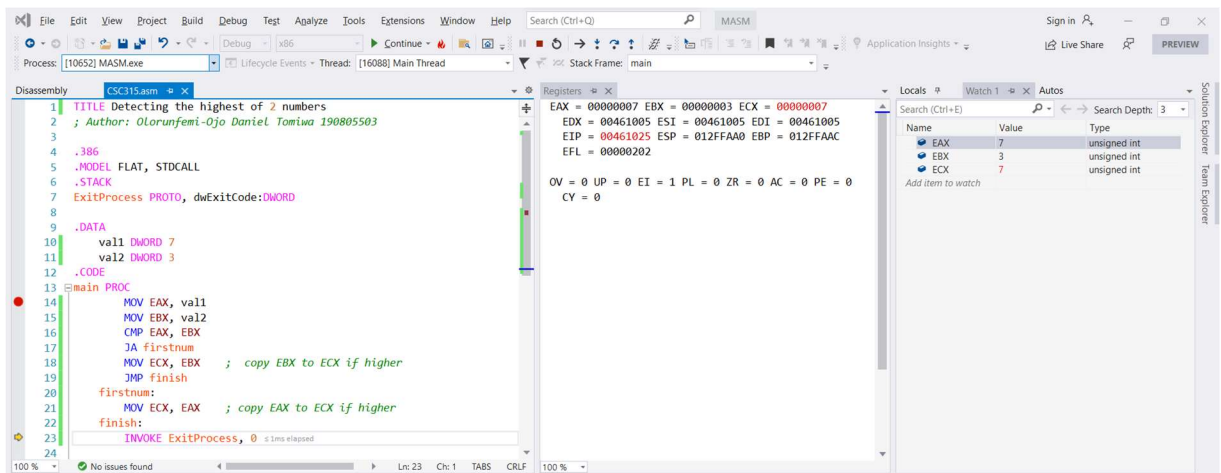
OV = 0 UP = 0 EI = 1 PL = 0 ZR = 0 AC = 0 PE = 0
CY = 0

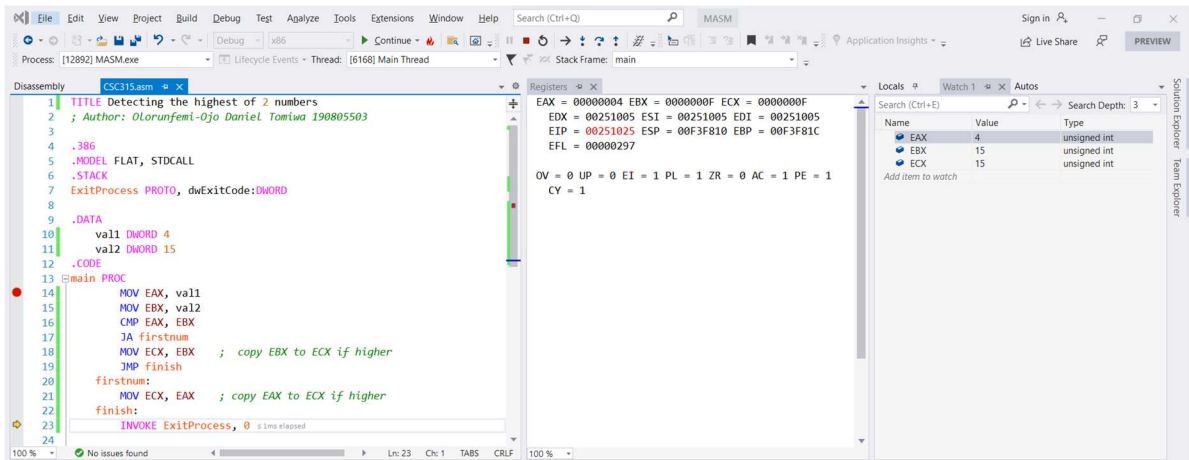
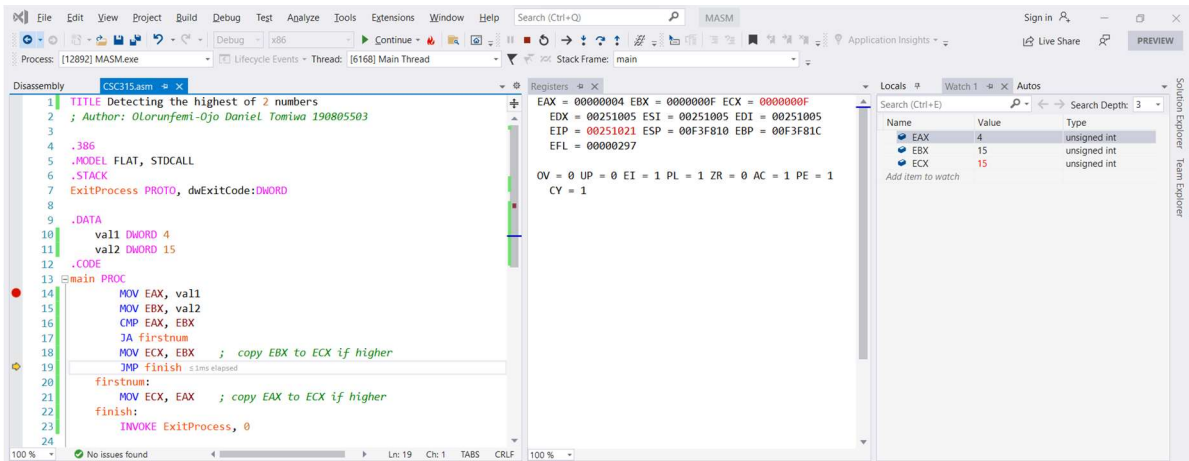
Locals:

Name	Value	Type
EAX	7	unsigned int
EBX	3	unsigned int
ECX	4591621	unsigned int

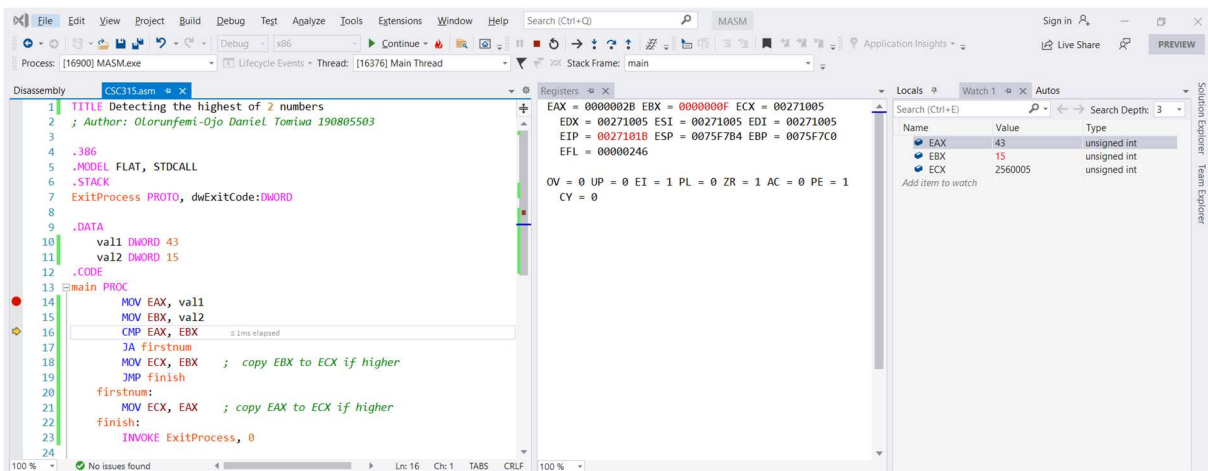


EXAMPLE 2: 4 and 15





EXAMPLE 3: 43 and 15



Disassembly: CSC315.asm

```

1  TITLE Detecting the highest of 2 numbers
2  ; Author: Olorunfemi-Ojo Daniel Tomiwa 190805503
3
4  .386
5  .MODEL FLAT, STDCALL
6  .STACK
7  ExitProcess PROTO, dwExitCode:DWORD
8
9  .DATA
10 val1 DWORD 43
11 val2 DWORD 15
12 .CODE
13 main PROC
14     MOV EAX, val1
15     MOV EBX, val2
16     CMP EAX, EBX
17     JA firstnum
18     MOV ECX, EBX ; copy EBX to ECX if higher
19     JMP finish
20 firstnum:
21     MOV ECX, EAX ; copy EAX to ECX if higher
22 finish:
23     INVOKE ExitProcess, 0
24

```

Registers:

EAX = 0000002B EBX = 0000000F ECX = 00271005
EDX = 00271005 ESI = 00271005 EDI = 00271005
EIP = 0027101D ESP = 0075F7B4 EBP = 0075F7C0
EFL = 00000212

OV = 0 UP = 0 EI = 1 PL = 0 ZR = 0 AC = 1 PE = 0
CY = 0

Locals:

Name	Value	Type
EAX	43	unsigned int
EBX	15	unsigned int
ECX	2560005	unsigned int

Disassembly: CSC315.asm

```

1  TITLE Detecting the highest of 2 numbers
2  ; Author: Olorunfemi-Ojo Daniel Tomiwa 190805503
3
4  .386
5  .MODEL FLAT, STDCALL
6  .STACK
7  ExitProcess PROTO, dwExitCode:DWORD
8
9  .DATA
10 val1 DWORD 43
11 val2 DWORD 15
12 .CODE
13 main PROC
14     MOV EAX, val1
15     MOV EBX, val2
16     CMP EAX, EBX
17     JA firstnum
18     MOV ECX, EBX ; copy EBX to ECX if higher
19     JMP finish
20 firstnum:
21     MOV ECX, EAX ; copy EAX to ECX if higher
22 finish:
23     INVOKE ExitProcess, 0
24

```

Registers:

EAX = 0000002B EBX = 0000000F ECX = 00271005
EDX = 00271005 ESI = 00271005 EDI = 00271005
EIP = 00271023 ESP = 0075F7B4 EBP = 0075F7C0
EFL = 00000212

OV = 0 UP = 0 EI = 1 PL = 0 ZR = 0 AC = 1 PE = 0
CY = 0

Locals:

Name	Value	Type
EAX	43	unsigned int
EBX	15	unsigned int
ECX	2560005	unsigned int

Disassembly: CSC315.asm

```

1  TITLE Detecting the highest of 2 numbers
2  ; Author: Olorunfemi-Ojo Daniel Tomiwa 190805503
3
4  .386
5  .MODEL FLAT, STDCALL
6  .STACK
7  ExitProcess PROTO, dwExitCode:DWORD
8
9  .DATA
10 val1 DWORD 43
11 val2 DWORD 15
12 .CODE
13 main PROC
14     MOV EAX, val1
15     MOV EBX, val2
16     CMP EAX, EBX
17     JA firstnum
18     MOV ECX, EBX ; copy EBX to ECX if higher
19     JMP finish
20 firstnum:
21     MOV ECX, EAX ; copy EAX to ECX if higher
22 finish:
23     INVOKE ExitProcess, 0
24

```

Registers:

EAX = 0000002B EBX = 0000000F ECX = 0000002B
EDX = 00271005 ESI = 00271005 EDI = 00271005
EIP = 00271025 ESP = 0075F7B4 EBP = 0075F7C0
EFL = 00000212

OV = 0 UP = 0 EI = 1 PL = 0 ZR = 0 AC = 1 PE = 0
CY = 0

Locals:

Name	Value	Type
EAX	43	unsigned int
EBX	15	unsigned int
ECX	43	unsigned int

EXAMPLE 4: 43 and 102

Disassembly: CSC315.asm

```
1  TITLE Detecting the highest of 2 numbers
2  ; Author: Olorunfemi-Ojo Daniel Tomiwa 190805503
3
4  .386
5  .MODEL FLAT, STDCALL
6  .STACK
7  ExitProcess PROTO, dwExitCode:DWORD
8
9  .DATA
10 val1 DWORD 43
11 val2 DWORD 102
12 .CODE
13 main PROC
14     MOV EAX, val1
15     MOV EBX, val2
16     CMP EAX, EBX
17     JA firstnum
18     MOV ECX, EBX ; copy EBX to ECX if higher
19     JMP finish
20 firstnum:
21     MOV ECX, EAX ; copy EAX to ECX if higher
22 finish:
23     INVOKE ExitProcess, 0
24
```

Registers:

EAX = 0000002B EBX = 00000066 ECX = 00ED1005
EDX = 00ED1005 ESI = 00ED1005 EDI = 00ED1005
EIP = 00ED101B ESP = 001EFA74 EBP = 001EFA80
EFL = 00000246

OV = 0 UP = 0 EI = 1 PL = 0 ZR = 1 AC = 0 PE = 1
CY = 0

Locals:

Name	Value	Type
EAX	43	unsigned int
EBX	102	unsigned int
ECX	15536133	unsigned int

Disassembly: CSC315.asm

```
1  TITLE Detecting the highest of 2 numbers
2  ; Author: Olorunfemi-Ojo Daniel Tomiwa 190805503
3
4  .386
5  .MODEL FLAT, STDCALL
6  .STACK
7  ExitProcess PROTO, dwExitCode:DWORD
8
9  .DATA
10 val1 DWORD 43
11 val2 DWORD 102
12 .CODE
13 main PROC
14     MOV EAX, val1
15     MOV EBX, val2
16     CMP EAX, EBX
17     JA firstnum
18     MOV ECX, EBX ; copy EBX to ECX if higher
19     JMP finish
20 firstnum:
21     MOV ECX, EAX ; copy EAX to ECX if higher
22 finish:
23     INVOKE ExitProcess, 0
24
```

Registers:

EAX = 0000002B EBX = 00000066 ECX = 00ED1005
EDX = 00ED1005 ESI = 00ED1005 EDI = 00ED1005
EIP = 00ED101F ESP = 001EFA74 EBP = 001EFA80
EFL = 00000287

OV = 0 UP = 0 EI = 1 PL = 1 ZR = 0 AC = 0 PE = 1
CY = 1

Locals:

Name	Value	Type
EAX	43	unsigned int
EBX	102	unsigned int
ECX	15536133	unsigned int

Disassembly: CSC315.asm

```
1  TITLE Detecting the highest of 2 numbers
2  ; Author: Olorunfemi-Ojo Daniel Tomiwa 190805503
3
4  .386
5  .MODEL FLAT, STDCALL
6  .STACK
7  ExitProcess PROTO, dwExitCode:DWORD
8
9  .DATA
10 val1 DWORD 43
11 val2 DWORD 102
12 .CODE
13 main PROC
14     MOV EAX, val1
15     MOV EBX, val2
16     CMP EAX, EBX
17     JA firstnum
18     MOV ECX, EBX ; copy EBX to ECX if higher
19     JMP finish
20 firstnum:
21     MOV ECX, EAX ; copy EAX to ECX if higher
22 finish:
23     INVOKE ExitProcess, 0
24
```

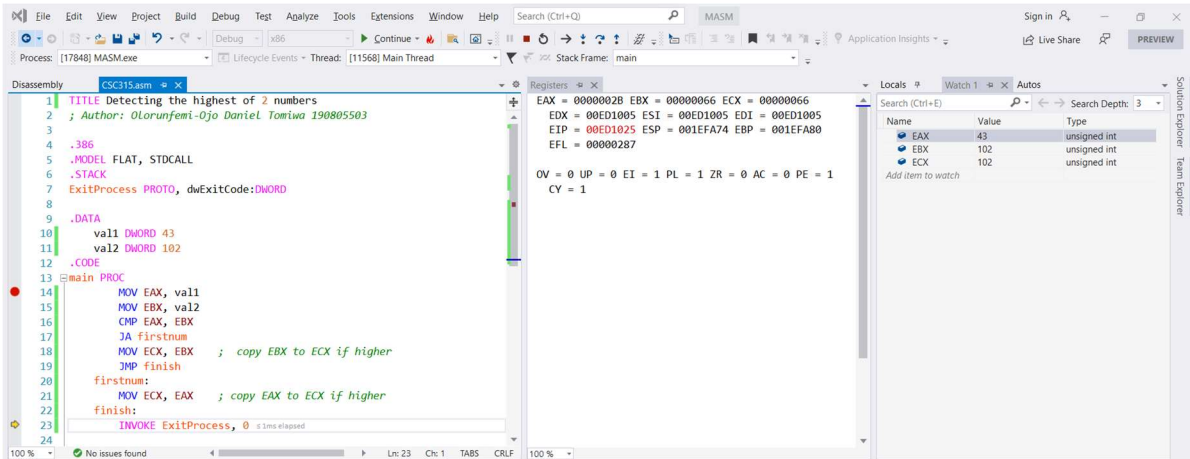
Registers:

EAX = 0000002B EBX = 00000066 ECX = 00000066
EDX = 00ED1005 ESI = 00ED1005 EDI = 00ED1005
EIP = 00ED1021 ESP = 001EFA74 EBP = 001EFA80
EFL = 00000287

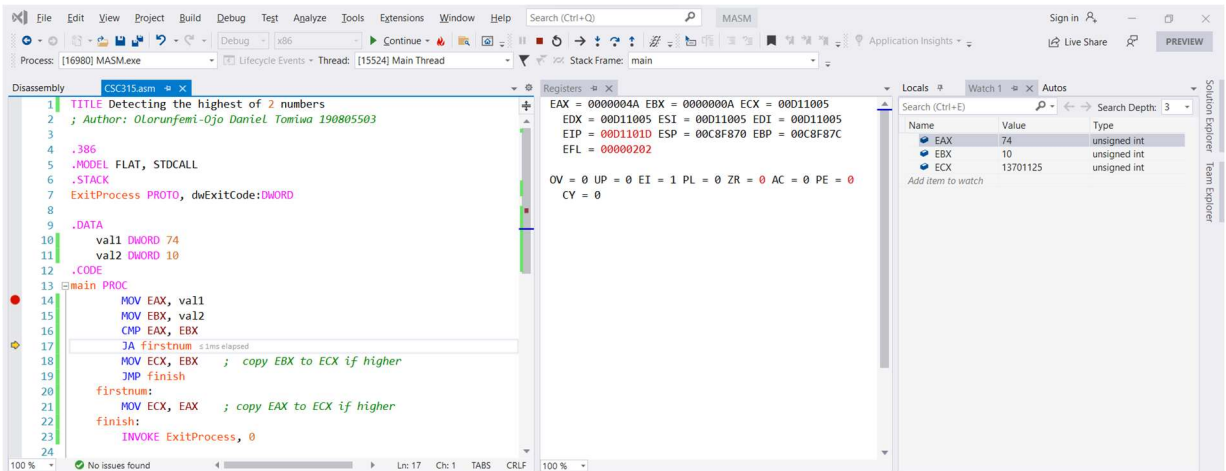
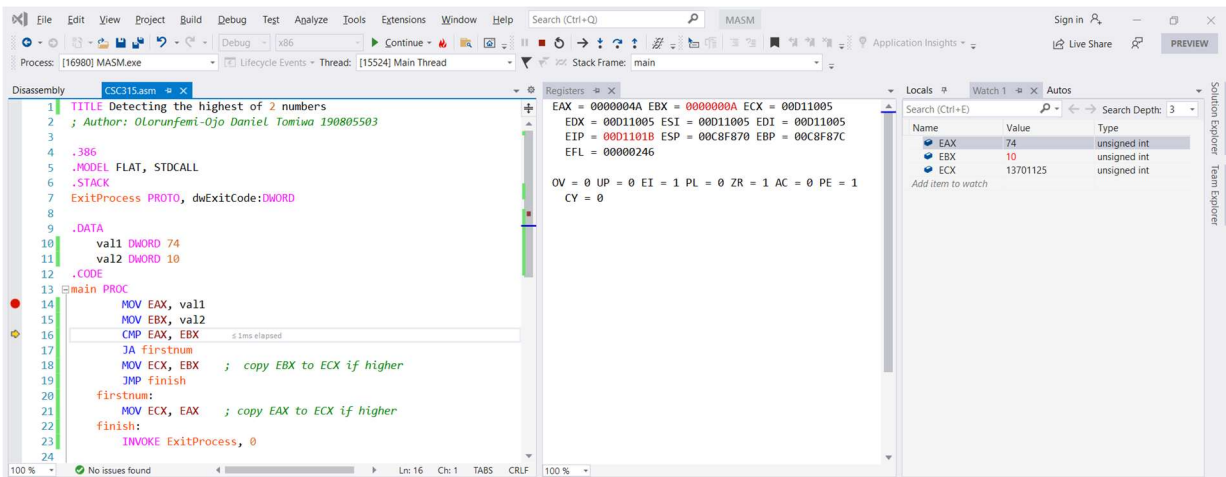
OV = 0 UP = 0 EI = 1 PL = 1 ZR = 0 AC = 0 PE = 1
CY = 1

Locals:

Name	Value	Type
EAX	43	unsigned int
EBX	102	unsigned int
ECX	102	unsigned int



EXAMPLE 5: 74 and 10



Visual Studio IDE showing the disassembly of a program. The assembly code is as follows:

```
1 TITLE Detecting the highest of 2 numbers
2 ; Author: Olurunfemi-Ojo Daniel Tomiwa 190805503
3
4 .386
5 .MODEL FLAT, STDCALL
6 .STACK
7 ExitProcess PROTO, dwExitCode:DWORD
8
9 .DATA
10 val1 DWORD 74
11 val2 DWORD 10
12 .CODE
13 main PROC
14     MOV EAX, val1
15     MOV EBX, val2
16     CMP EAX, EBX
17     JA firstnum
18     MOV ECX, EBX ; copy EBX to ECX if higher
19     JMP finish
20 firstnum:
21     MOV ECX, EAX ; copy EAX to ECX if higher
22 finish:
23     INVOKE ExitProcess, 0
24
```

The Registers window shows the following values:

Register	Value	Type
EAX	0000004A	unsigned int
EBX	0000000A	unsigned int
ECX	00D11005	unsigned int
EDX	00D11005	unsigned int
ESI	00D11005	unsigned int
EDI	00D11005	unsigned int
EIP	00D11023	unsigned int
ESP	00C8F870	unsigned int
EBP	00C8F87C	unsigned int
EFL	00000202	unsigned int

The Locals window shows the following values:

Name	Value	Type
EAX	74	unsigned int
EBX	10	unsigned int
ECX	13701125	unsigned int

Visual Studio IDE showing the disassembly of a program. The assembly code is as follows:

```
1 TITLE Detecting the highest of 2 numbers
2 ; Author: Olurunfemi-Ojo Daniel Tomiwa 190805503
3
4 .386
5 .MODEL FLAT, STDCALL
6 .STACK
7 ExitProcess PROTO, dwExitCode:DWORD
8
9 .DATA
10 val1 DWORD 74
11 val2 DWORD 10
12 .CODE
13 main PROC
14     MOV EAX, val1
15     MOV EBX, val2
16     CMP EAX, EBX
17     JA firstnum
18     MOV ECX, EBX ; copy EBX to ECX if higher
19     JMP finish
20 firstnum:
21     MOV ECX, EAX ; copy EAX to ECX if higher
22 finish:
23     INVOKE ExitProcess, 0
24
```

The Registers window shows the following values:

Register	Value	Type
EAX	0000004A	unsigned int
EBX	0000000A	unsigned int
ECX	0000004A	unsigned int
EDX	00D11005	unsigned int
ESI	00D11005	unsigned int
EDI	00D11005	unsigned int
EIP	00D11023	unsigned int
ESP	00C8F870	unsigned int
EBP	00C8F87C	unsigned int
EFL	00000202	unsigned int

The Locals window shows the following values:

Name	Value	Type
EAX	74	unsigned int
EBX	10	unsigned int
ECX	74	unsigned int