

LAB TASK # 08

MOHAMMAD BASIL ALI KHAN

20K-0477

Example:

The screenshot shows the Visual Studio IDE with the assembly file `Text.asm` open. The code includes `Irvine32.inc` and defines a `main` procedure. The assembly code is as follows:

```
1 INCLUDE Irvine32.inc
2
3 .code
4 main PROC
5     mov al, 10101110b    ; Clear only bit 3
6     and al, 11110110b    ; AL = 10100110
7     mov al, 11100011b    ; set bit 2
8     or al, 00000100b     ; AL = 11100111
9     mov al, 10110101b    ; 5 bits means odd parity
10    xor al, 0             ; PF=0(P0)
11    mov al, 10100101b    ; 4 bits means even parity
12    xor al, 0             ; PF=1(PF)
13    mov al, 11110000b    ; AL = 00001111b
14    not al                ; AL = 00001111b
15    mov al, 00100101b    ; ZF=0
16    test al, 00001001b   ; ZF=0
17    mov al, 00100101b    ; ZF=1
18    test al, 00001000b   ; ZF=1
19    call DumpRegs
20    exit
21 main ENDP
22 END main
```

The Microsoft Visual Studio Debug Console shows the following output:

```
EAX=001EF925 EBX=00254000 ECX=0007100A EDX=0007100A
ESI=0007100A EDI=0007100A EBP=001EF92C ESP=001EF920
EIP=00073681 EFL=00000246 CF=0 SF=0 ZF=1 OF=0 AF=0 PF=1

C:\Users\student\source\repos\Project7\Debug\Project7.exe (process 72856) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically
close the console when debugging stops.
Press any key to close this window . . .
```

Example:

The screenshot shows the Visual Studio IDE with the assembly file `Text.asm` open. The code includes `Irvine32.inc` and defines a `main` procedure. The assembly code is as follows:

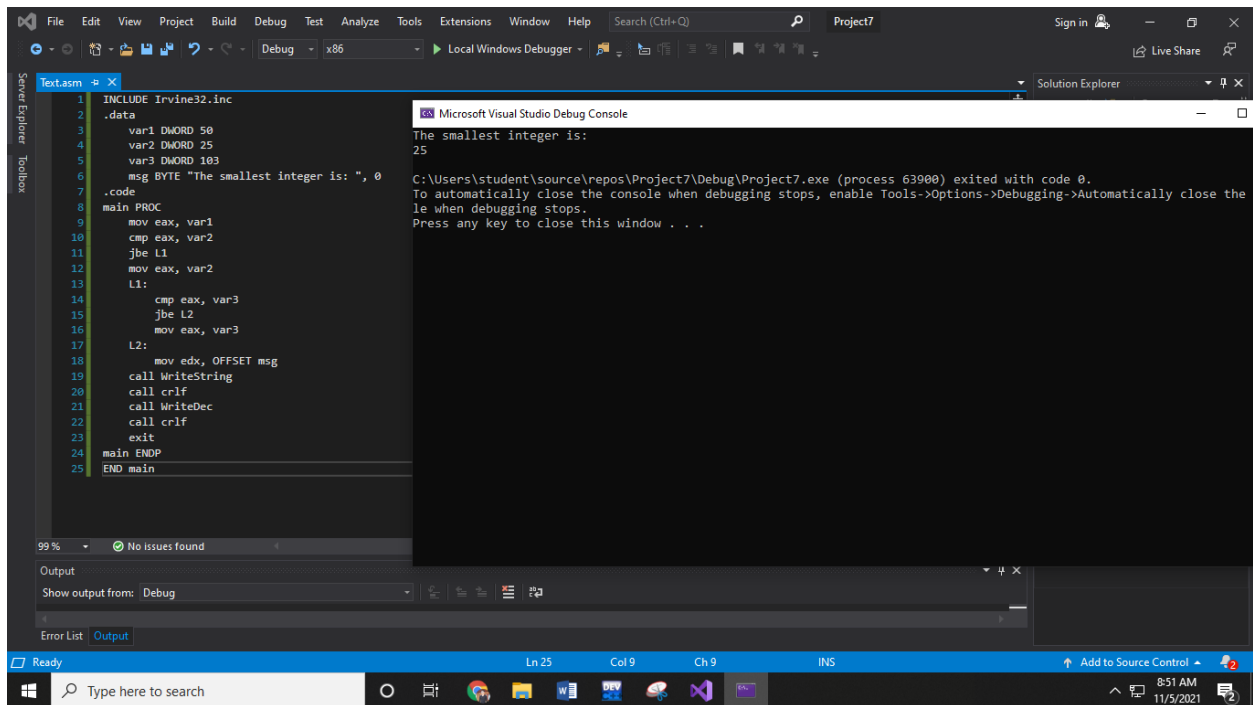
```
1 INCLUDE Irvine32.inc
2 .data
3     var1 DWORD 500
4     var2 DWORD 125
5     larger DWORD ?
6 .code
7 main PROC
8     mov eax, var1
9     mov larger, eax
10    mov ebx, var2
11    cmp eax, ebx
12    jae L1
13    mov larger, ebx
14    L1:
15    call Writedec
16    exit
17 main ENDP
18 END main
```

The Microsoft Visual Studio Debug Console shows the following output:

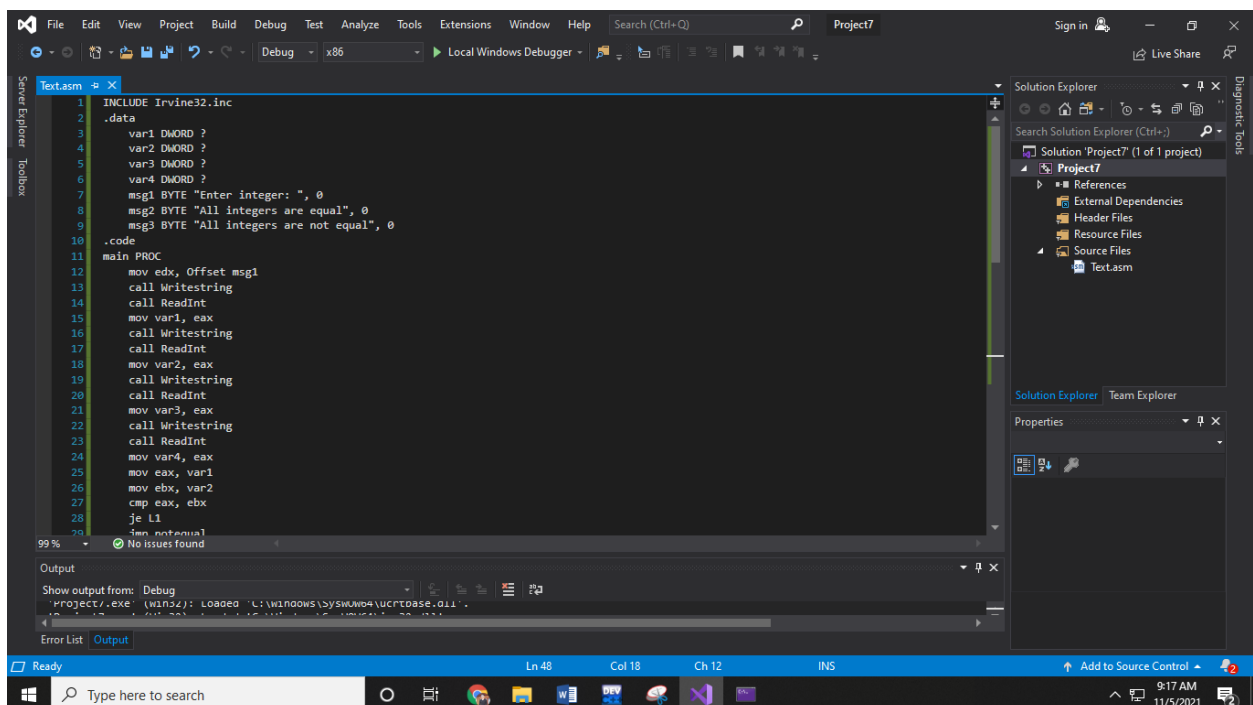
```
500

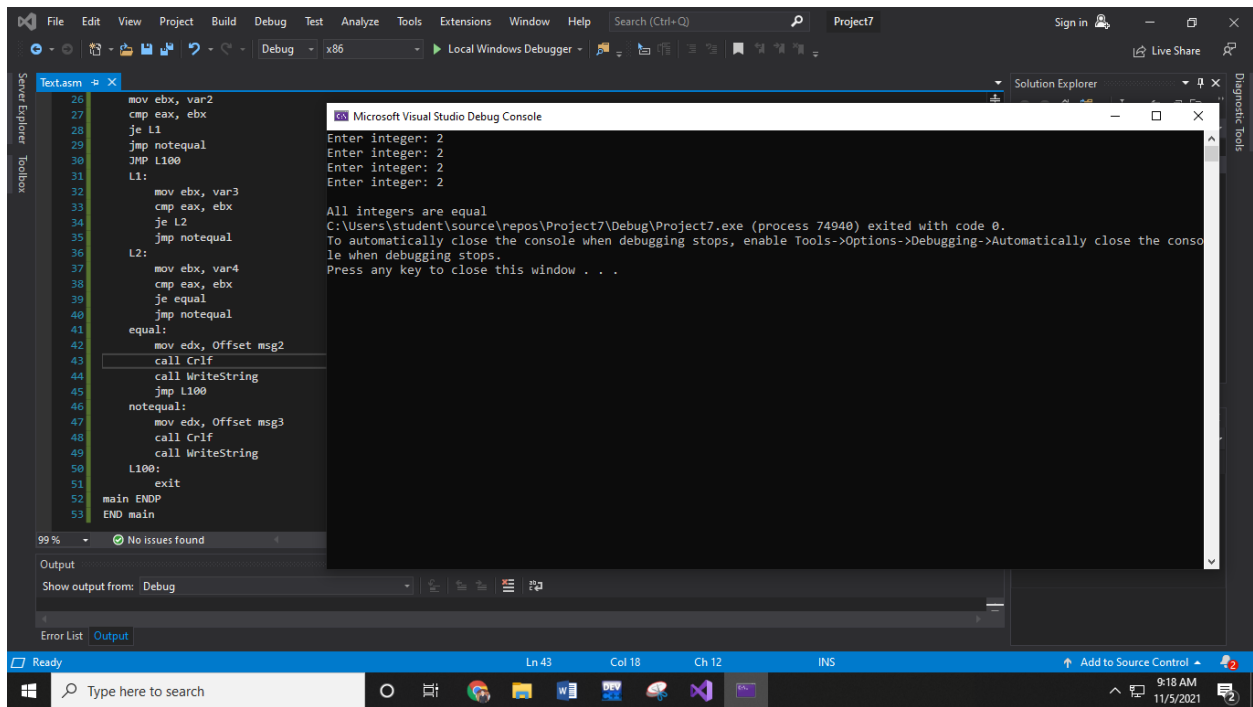
C:\Users\student\source\repos\Project7\Debug\Project7.exe (process 1508) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console
when debugging stops.
Press any key to close this window . . .
```

Example:

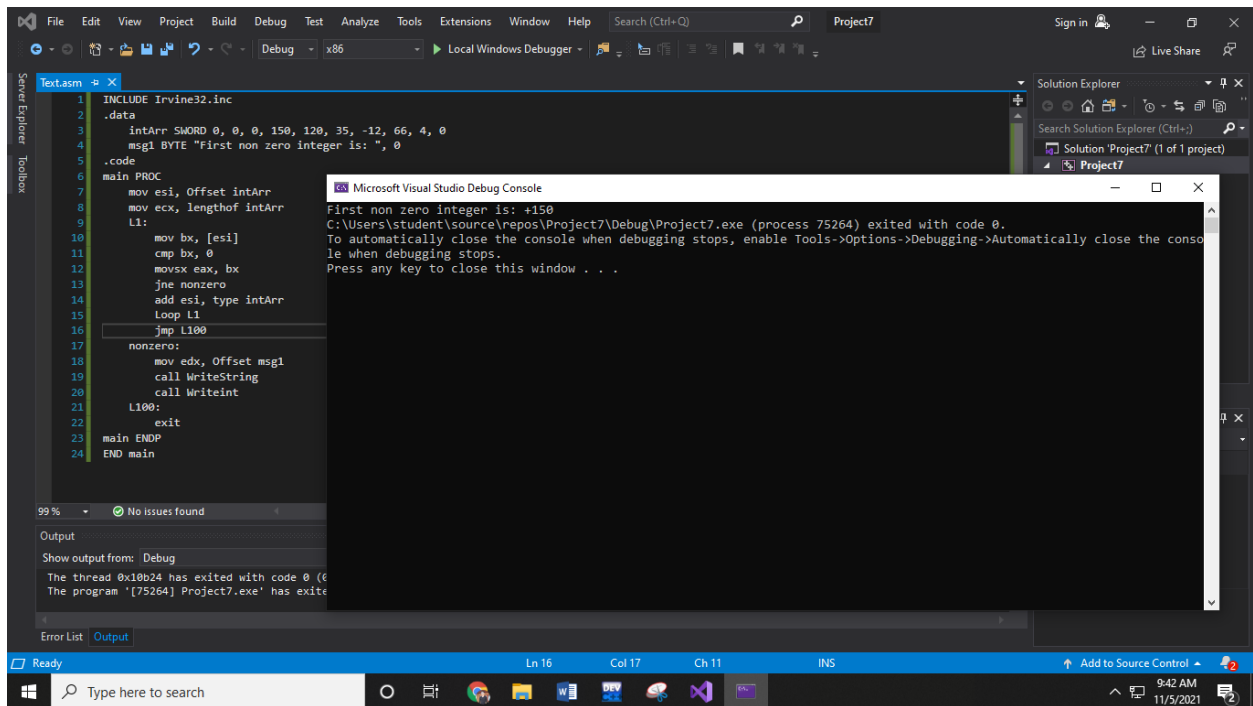


Task#01

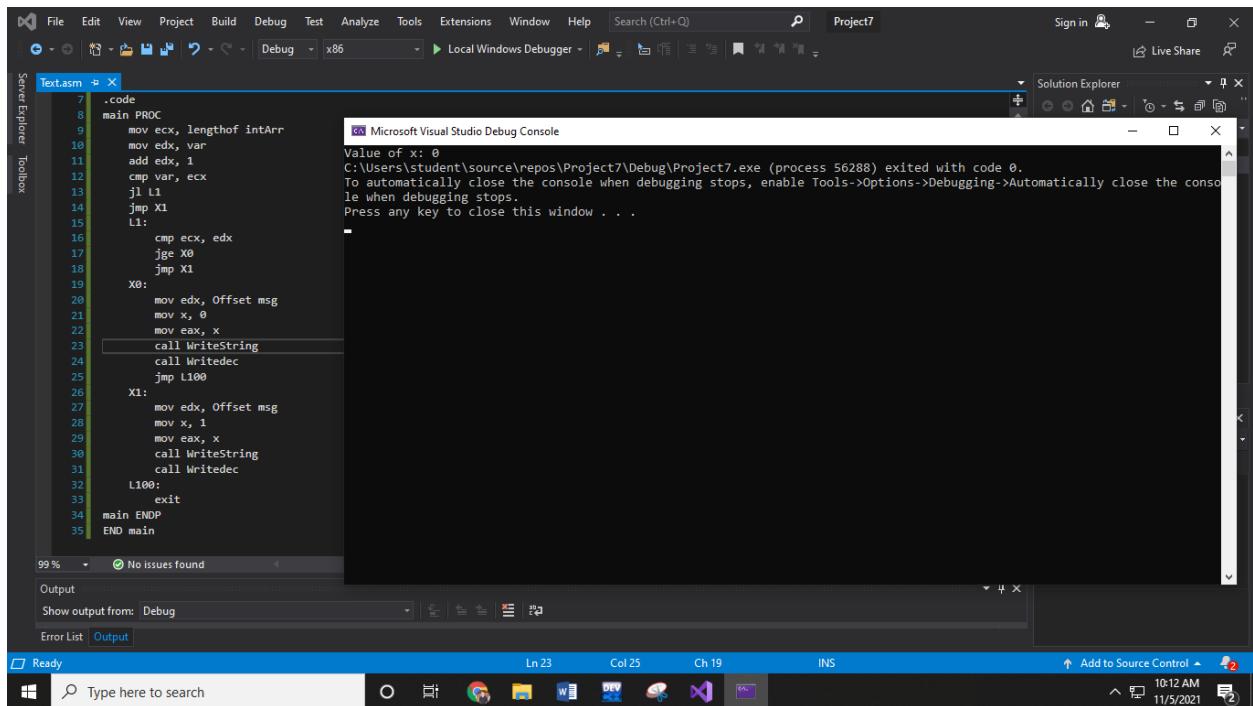
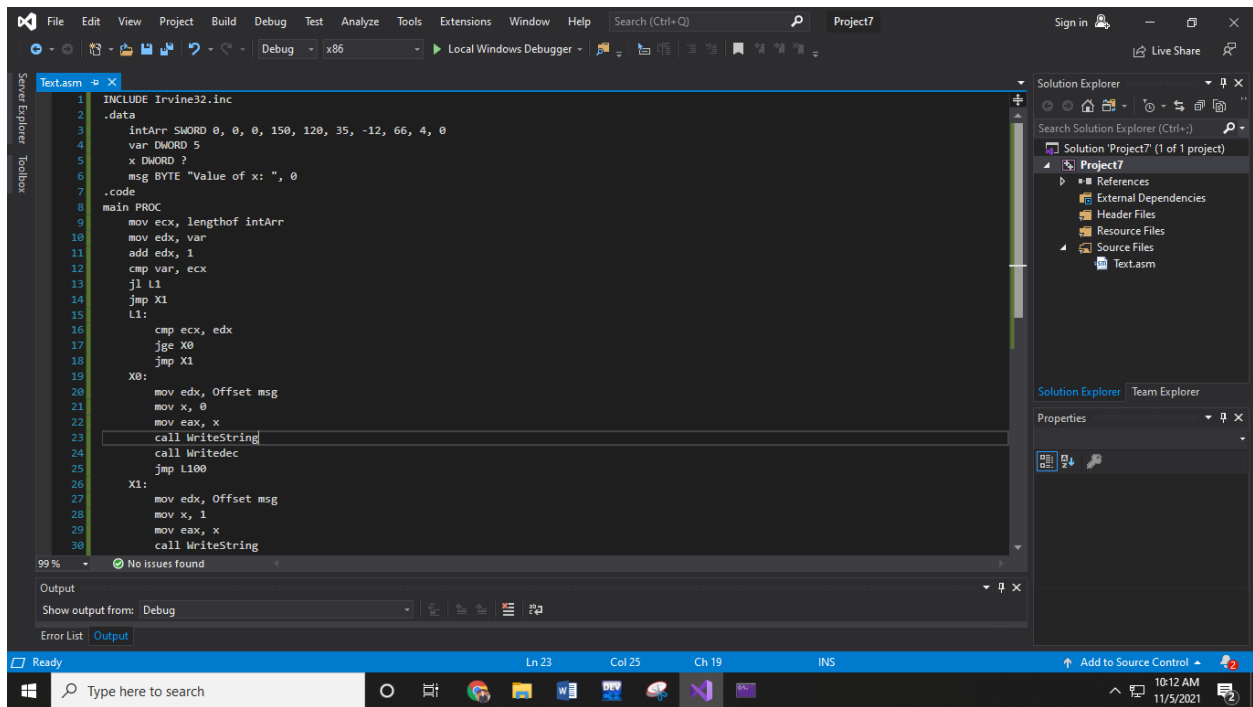




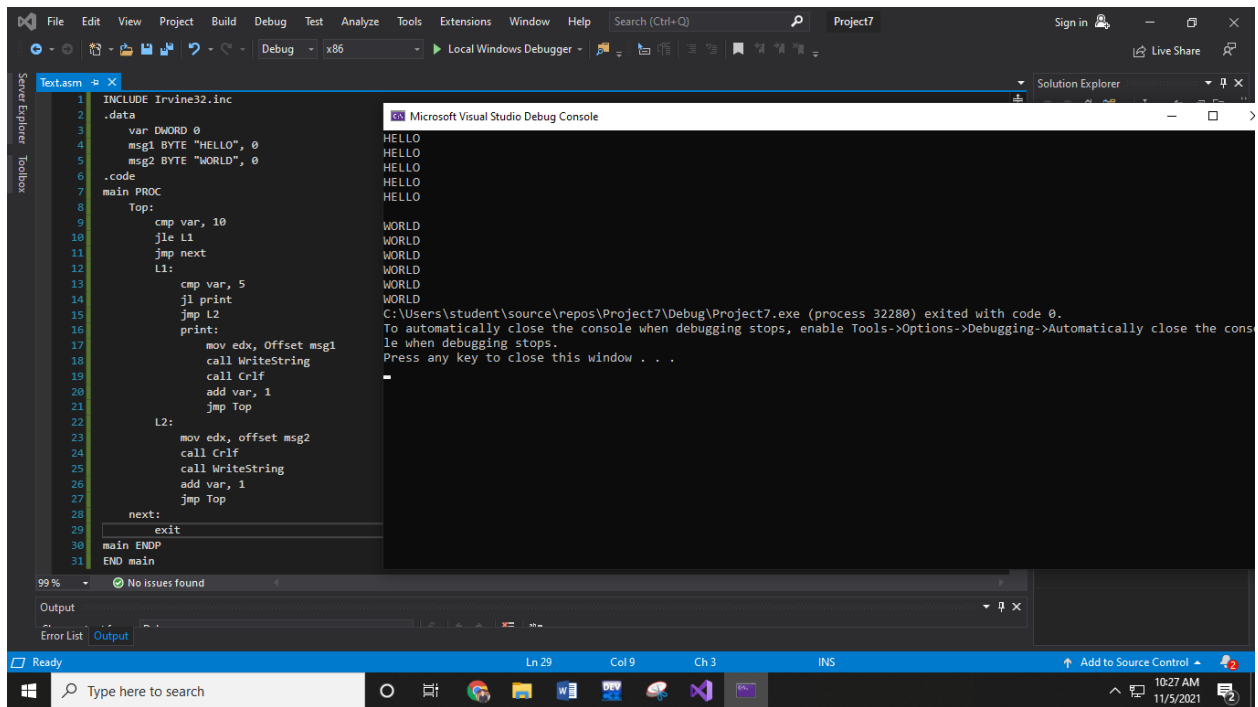
Task#02



Task#03:



Task#04:

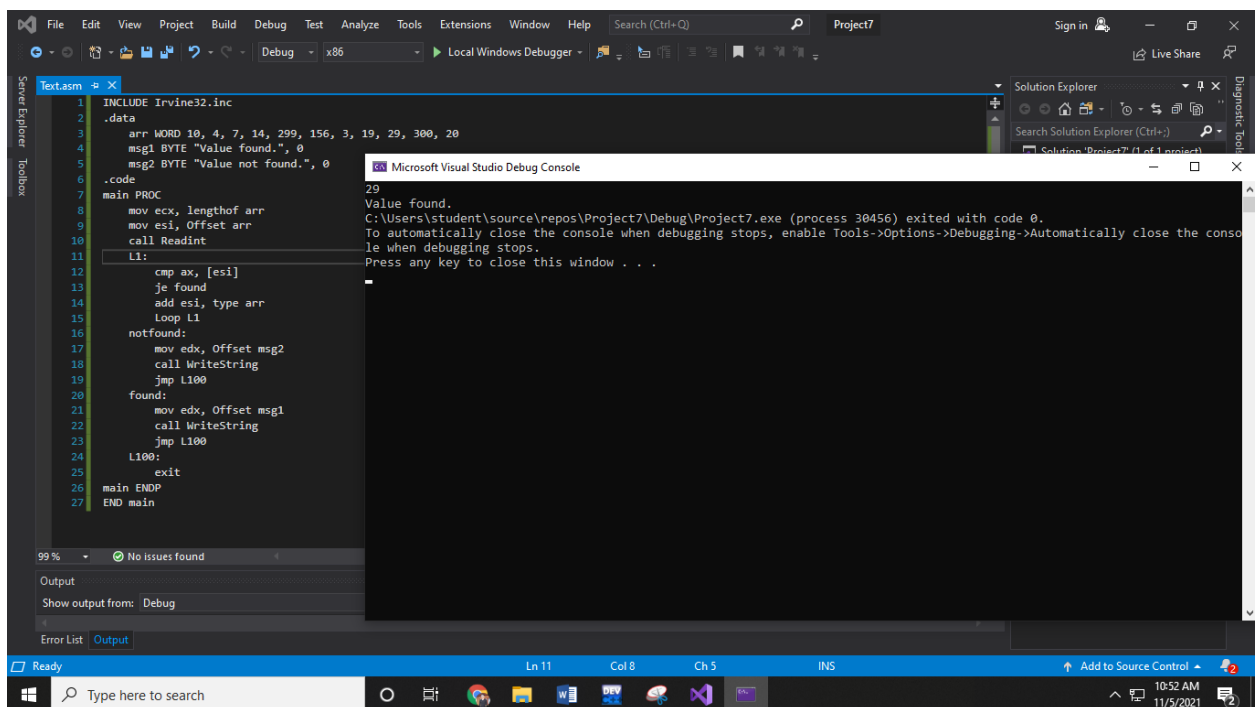


```
1 INCLUDE Irvine32.inc
2 .data
3     var DWORD 0
4     msg1 BYTE "HELLO", 0
5     msg2 BYTE "WORLD", 0
6 .code
7 main PROC
8     Top:
9     cmp var, 10
10    jle L1
11    jmp next
12    L1:
13    cmp var, 5
14    jl print
15    jmp L2
16    print:
17    mov edx, Offset msg1
18    call WriteString
19    call Crlf
20    add var, 1
21    jmp Top
22    L2:
23    mov edx, Offset msg2
24    call Crlf
25    call WriteString
26    add var, 1
27    jmp Top
28    next:
29    exit
30 main ENDP
31 END main
```

Microsoft Visual Studio Debug Console

```
HELLO
HELLO
HELLO
HELLO
HELLO
WORLD
WORLD
WORLD
WORLD
WORLD
C:\Users\student\source\repos\Project7\Debug\Project7.exe (process 32280) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```

Task#05:



```
1 INCLUDE Irvine32.inc
2 .data
3     arr WORD 10, 4, 7, 14, 299, 156, 3, 19, 29, 300, 20
4     msg1 BYTE "Value found.", 0
5     msg2 BYTE "Value not found.", 0
6 .code
7 main PROC
8     mov ecx, lengthof arr
9     mov esi, Offset arr
10    call ReadInt
11    L1:
12    cmp ax, [esi]
13    je found
14    add esi, type arr
15    loop L1
16    notfound:
17    mov edx, Offset msg2
18    call WriteString
19    jmp L100
20    found:
21    mov edx, Offset msg1
22    call WriteString
23    jmp L100
24    L100:
25    exit
26 main ENDP
27 END main
```

Microsoft Visual Studio Debug Console

```
29
Value found.
C:\Users\student\source\repos\Project7\Debug\Project7.exe (process 30456) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```

Task#06:

The screenshot shows the Visual Studio Code editor with a project named 'COAL LAB 08'. The active file is 'Text.asm', which contains the following assembly code:

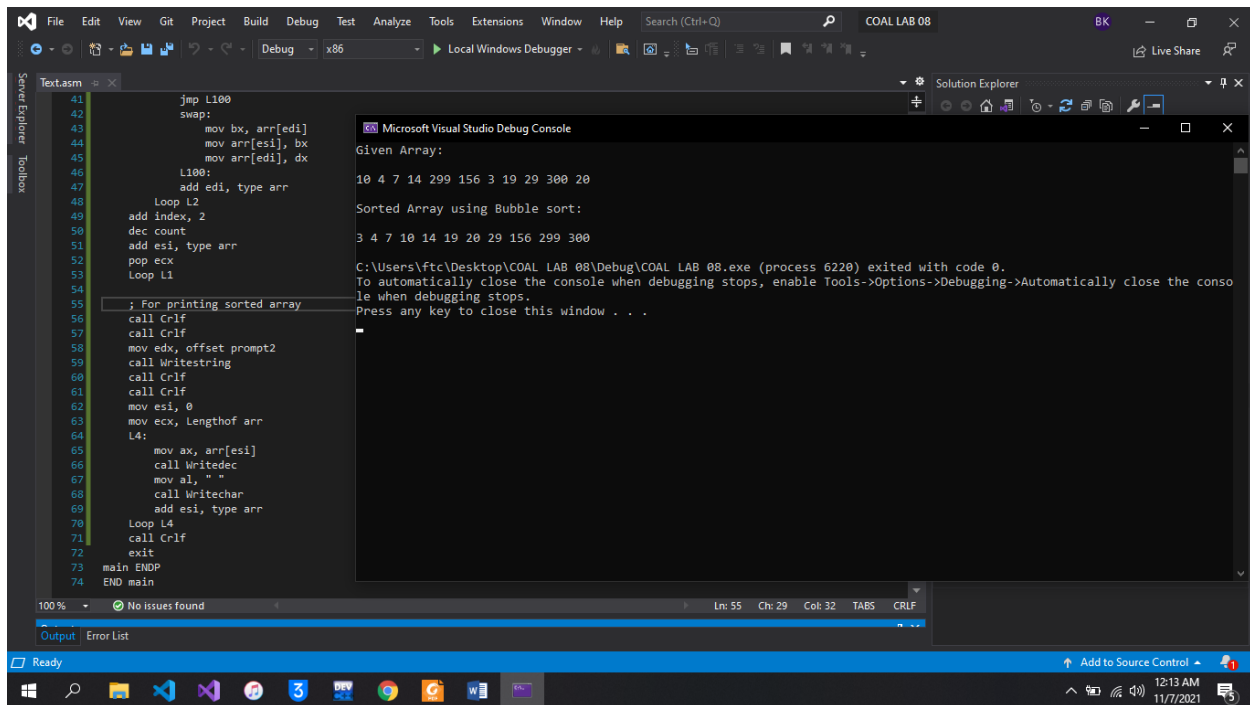
```
1 INCLUDE Irvine32.inc
2
3 .data
4 arr WORD 10, 4, 7, 14, 299, 156, 3, 19, 29, 300, 20
5 count dword 11
6 index dword 0
7 prompt1 BYTE "Given Array: ",0
8 prompt2 BYTE "Sorted Array using Bubble sort: ",0
9 .code
10 main PROC
11     ; For printing given array
12     mov edx, offset prompt1
13     call WriteString
14     call CrLf
15     call CrLf
16     mov eax, 0
17     mov esi, 0
18     mov ecx, Lengthof arr
19     L3:
20     mov ax, arr[esi]
21     call WriteDec
22     mov al, " "
23     call WriteChar
24     add esi, type arr
25     Loop L3
26
27     ; Method for bubble sort
28     mov ebx, 0
29     mov edx, 0
30     mov eax, 0
31     mov esi, 0
32     mov ecx, lengthof arr
33     L1:
34     push ecx
35     mov ecx, count
```

The Solution Explorer on the right shows the project structure: 'Solution 'COAL LAB 08' (1 of 1 project)' containing 'References', 'External Dependencies', 'Header Files', 'Resource Files', 'Source Files', and 'Text.asm'. The Properties window is empty.

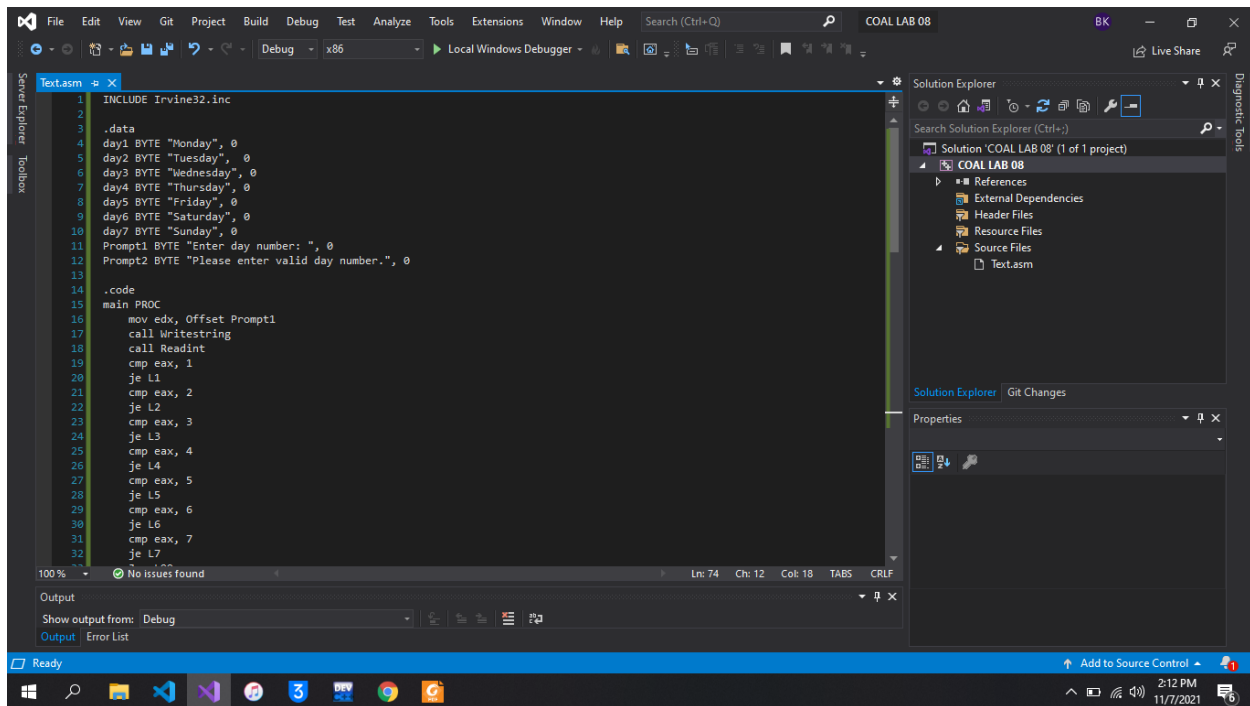
The screenshot shows the Visual Studio Code editor with the same project 'COAL LAB 08'. The 'Text.asm' file now contains the completed assembly code for a bubble sort program:

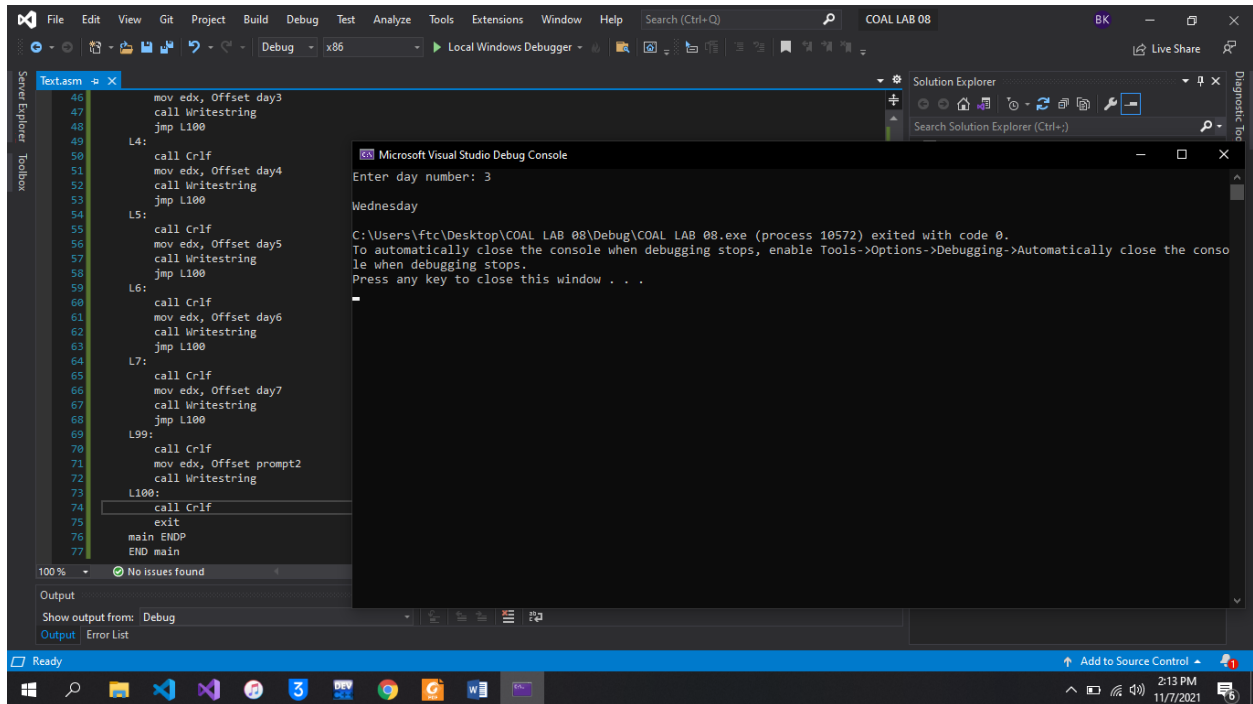
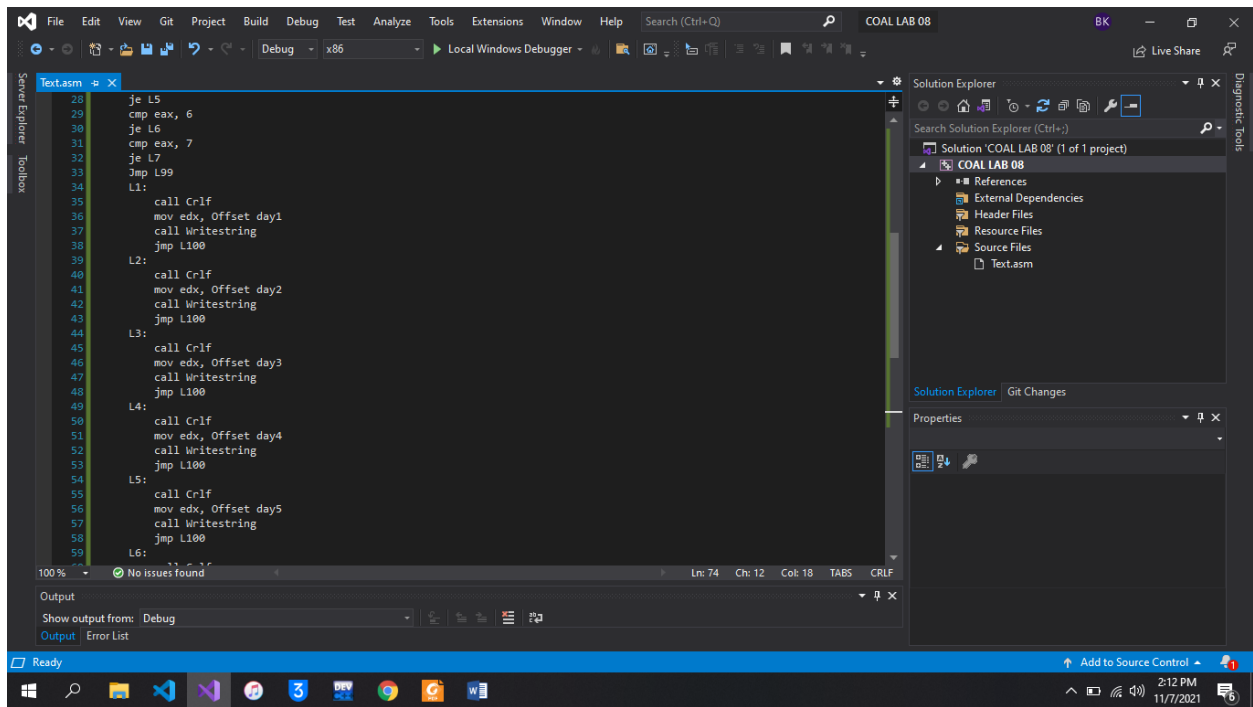
```
26
27 ; Method for bubble sort
28 mov ebx, 0
29 mov edx, 0
30 mov eax, 0
31 mov esi, 0
32 mov ecx, lengthof arr
33 L1:
34     push ecx
35     mov ecx, count
36     mov edi, index
37     L2:
38     mov dx, arr[esi]
39     cmp dx, arr[edi]
40     jg swap
41     jmp L100
42 swap:
43     mov bx, arr[edi]
44     mov arr[esi], dx
45     mov arr[edi], bx
46     L100:
47     add edi, type arr
48     Loop L2
49     add index, 2
50     dec count
51     add esi, type arr
52     pop ecx
53     Loop L1
54
55 ; For printing sorted array
56 call CrLf
57 call CrLf
58 mov edx, offset prompt2
59 call WriteString
60 call CrLf
```

The Solution Explorer and Properties window remain the same as in the first screenshot.



Task#07:





Task#08:

