

Question 1:

```
Registers
EAX = 00000000 EBX = 0083D000 ECX = 00000000 EDX = 00A71005 ESI = 0000001E EDI = 0000003C EIP = 00A71034 ESP = 00B7FB90 EBP = 00B7FB9C EFL = 00000206

160%
Autos Locals Registers Threads Modules Watch 1
Solution Explorer
Search Solution Explorer (Ctrl+...)
Solution 'Lab3' (1 of 1 project)
  Lab3
    References
    External Dependencies
    Header Files
    Resource Files
    Source Files
      Jacob_Graham_1141178_LAB3_Question1.asm

13 main PROC
14     mov esi, 0          ; set up source index
15     mov edi, 0          ; set up destination index
16     mov ecx, wArraySize ; set up loop counter
17
18 loopThroughArray:
19     movzx eax, wArray[esi] ; Load word into eax
20     mov dword ptr dArray[edi], eax ; store word in doubleword array
21     add esi, 2            ; increment source index by 2 bytes (1 word)
22     add edi, 4            ; increment destination index by 4 bytes (1 dword)
23     loop loopThroughArray ; decrement counter and loop if not zero
24
25     INVOKE ExitProcess, 0 < 1ms elapsed
26 main ENDP
27 END main
28
```

Question 2:

```
Registers
EAX = 00000000 EBX = 00ECC000 ECX = 00000000 EDX = 00201005 ESI = 00000007 EDI = 00201005 EIP = 00201042 ESP = 010FFC98 EBP = 010FFCA4 EFL = 00000202

60%
Autos Locals Registers Threads Modules Watch 1
Solution Explorer
Search Solution Explorer (Ctrl+...)
Solution 'Lab3' (1 of 1 project)
  Lab3
    References
    External Dependencies
    Header Files
    Resource Files
    Source Files
      Jacob_Graham_1141178_LAB3_Question2.asm

16 add ebx, [fib + esi - 2] ; Add two previous values
17 mov [fib + esi], ebx ; Store result
18 inc esi ; Move to next array element
19 loop calculateFibValue
20 mov ecx, 7 ; Display the result in registers
21 mov esi, 0
22
23 reverseCharOrder:
24     mov ebx, [fib + esi] ; Get current value
25     inc esi
26     loop reverseCharOrder
27     INVOKE ExitProcess, 0 < 1ms elapsed
28
29 main ENDP
30 END main
```