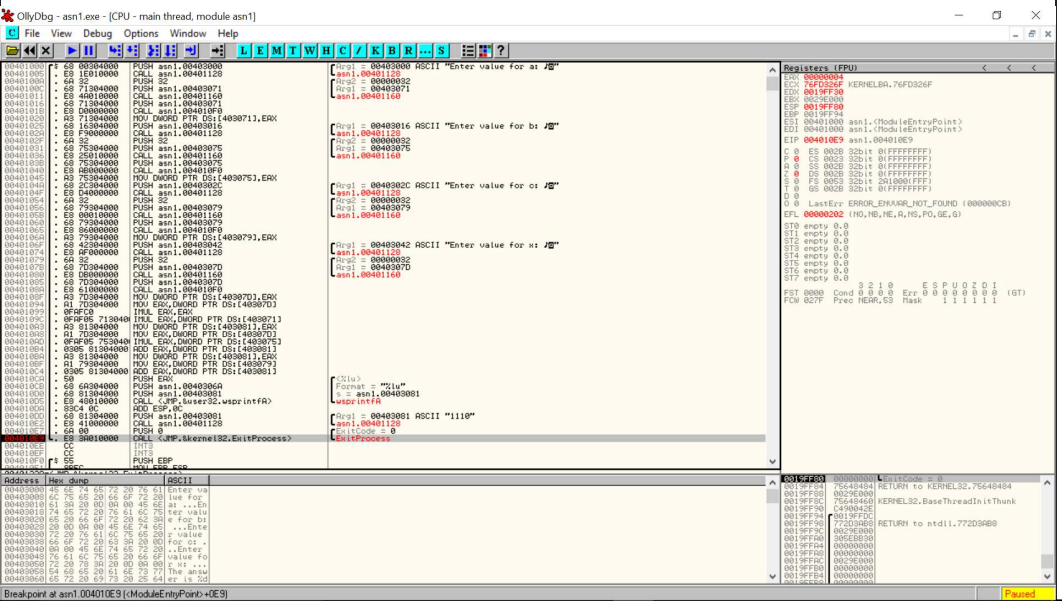
```
; CS 371 Assignment 1
; Ben Pazienza
; This file contains a program computes a*x^2 + b*x + c using values
; from the data segment



        include 371-prologue.inc                ; suck in standard prologue



        .const
    BUFFER_LENGTH equ 50
        .data
    a_msg byte "Enter value for a: ", CR, LF, 0
    b_msg byte "Enter value for b: ", CR, LF, 0
    c_msg byte "Enter value for c: ", CR, LF, 0
    x_msg byte "Enter value for x: ", CR, LF, 0
    msg byte "The answer is %d", 10, 0
    unsigned_integer_format byte "%lu", 0           ;doubleword to ASCII

        .data?
    buffer byte ?
    a dword ?
    b dword ?
    c1 dword ?
    x dword ?

    result dword ?

        .code
main    proc

        push offset a_msg        ;prompt user for value for a
        call StdOut
        push BUFFER_LENGTH
        push offset a
        call StdIn
        push offset a
        call atodw
        mov a, eax


        push offset b_msg        ;prompt user for value for b
        call StdOut
        push BUFFER_LENGTH
        push offset b
        call StdIn
        push offset b
        call atodw
        mov b, eax


        push offset c_msg        ;prompt user for value for c
        call StdOut
        push BUFFER_LENGTH
        push offset c1
        call StdIn
        push offset c1
        call atodw
        mov c1, eax
```



```
        push offset x_msg        ;prompt user for value for x
        call StdOut
        push BUFFER_LENGTH
        push offset x
        call StdIn
        push offset x
        call atodw
        mov x, eax

        mov eax, x       ; calculate ax^2 (stores in eax)
        imul eax, eax
        imul eax, a
        mov result, eax

        mov eax, x       ; calculates bx (adds to eax)
        imul eax, b
        add eax, result
        mov result, eax

        mov eax, c1       ; adds c to ax^2+bx
        add eax, result

        push eax
        push offset unsigned_integer_format
        push offset result
        call wsprintf
        add esp, 12

        push offset result
        call StdOut

        push 0                                  ; alternate standard exit
        call ExitProcess                        ; using library exit call
main    endp
        end  main
```

```
;EIP before: 00401000
;EIP after:  004010E9
;This difference could be because Memory locations 00401000 - 004010E9 were
;used in the process of running this program
```


OllyDbg - asn1.exe - [CPU - main thread, module asn1]
File View Debug Options Window Help
Registers (FPU)
Address Hex dump ASCII
Breakpoint at asn1.004010E9 (<ModuleEntryPoint>+0E9)
Paused