Lab 6 (50 points)

Q1- (25 points) Consider the following program

Data segment:

|  |  |  |
| --- | --- | --- |
| **var1** | **dd** | **179** |
| **var2** | **db** | **0A3h, 017h, 012h** |
| **var3** | **db** | **“bca”** |
|  |  |  |

Code:

**mov eax, var1 // move var1memory location to eax**

**add eax, 3 // add 3 to the memory location, i.e move 3 bytes from old location**

**mov ebx, [eax] // move 4 bytes starting at the new memory location to ebx, flip it first**

**add ebx, 5 // add 5 to the value stored in ebx.**

**mov [var1], ebx // move the new values in ebx to the memory location of var 1**

Show the contents of memory and registers before and after each instruction. Then show the final layout of the below memory starting at address “var1” on a Little-Endian Machine.

var1 var2 var3

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| B3 | 00 | 00 | 00 | A3 | 17 | 12 | 62 | 63 | 61 |

mov eax, var1

eax = FF FF 00 00 00

add eax, 3

eax = FF FF 00 00 03

mov ebx, [eax]

ebx = 12 17 A3 00

add ebx, 5

ebx = 17 17 A3 00

mov [var1], ebx

var1 = 00 A3 17 17

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 00 | A3 | 17 | 17 | A3 | 17 | 12 | 62 | 63 | 61 |

Var1 var2 var3

Q2- (25 points) Consider the following program

Data segment:

|  |  |  |
| --- | --- | --- |
| **var1** | **db** | **“b, “ca”, 0** |
| **var2** | **db** | **3, 0, 0, 0** |
| **var3 times 2** | **dw** | **012h** |

Code:

**mov eax, var3 // move var3 memory location to eax**

**mov ebx, var1 // move var1 memory location to ebx**

**sub eax, 4 // reduce 4 from memory location stored in eax**

**add ebx, [eax] // add the content of 4 bytes from new memory location to var1 (stored in ebx)**

// **New ebx = old ebx + 3, result, move 3 bytes from old ebx memory location**

**mov dword [ebx], 42 // 42d = 002Ah. Flip it, and save it at two bytes starting from memory**

**//location stored in ebx**

Show the contents of memory and registers before and after each instruction. Then show the final layout of the below memory starting at address “var1” on a Little-Endian Machine.

var1 var2 var3

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 62 | 63 | 61 | 0 | 3 | 0 | 0 | 0 | 0 | 0 | 12 | 12 |

mov eax, var3

eax = FF FF 00 08

mov ebx, var1

ebx = FF FF 00 00

sub eax, 4

eax = FF FF 00 04

add ebx, [eax]

ebx = 3 0 0 0

mov dword [ebx], 42

ebx = 0 3 2A 00

var1 var2 var3

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 62 | 63 | 61 | 0 | 3 | 0 | 0 | 0 | 0 | 0 | 12 | 12 |

(unsure if any of the variable data would change since just the registers are being altered)