**Computer Organization and Assembly Language**

Question #1:

Write instruction that set the Zero flag if the 32-bit value in EAX is even and clear the Zero flag if EAX is odd.

**Solution:**

TEST EAX, 1

Question #2:

Write a program convert the string into capital but only the lower-case characters.(i.e) Before conversion it should check the character is not already capital.

**Solution:**

include irvine32.inc

.data

msg BYTE "JAHaNzeb NAeem",0

msg2 BYTE "Before capitalization: ",0

msg3 BYTE "After capitalization: ",0

.code

main proc

call crlf

mov edx, OFFSET msg2

call WriteString

mov edx, OFFSET msg

call WriteString

mov esi, OFFSET msg

mov ecx, LENGTHOF msg

L1:

mov eax, 0

mov al, [esi]

cmp al, "a" ;ASCII values of capital letters is lower than small letters. So, we can skip letters with lower value than 'a'

JB skip

sub al, 32

mov [esi], al

skip:

inc esi

loop L1

call crlf

call crlf

mov edx, OFFSET msg3

call WriteString

mov edx, OFFSET msg

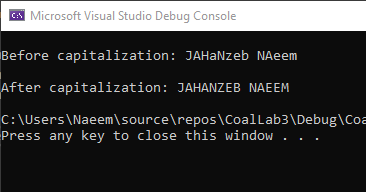
call WriteString

call crlf

exit

main endp

end main



Question#3:

Write a program that copies only unsigned numbers from Array1 to another Array2. The Array1 has both signed and unsigned numbers.

**Solution:**

include irvine32.inc

.data

mixedArray SDWORD -3,5,7,6,-2,3,4,9,0,-5,-3,4,5,0,8

array2 SDWORD 15 DUP (0)

msg BYTE "Original Array: ",0

msg2 BYTE "Unsigned Numbers: ",0

.code

main proc

call crlf

mov edx, OFFSET msg

call WriteString

mov esi, OFFSET mixedArray

mov ecx, LENGTHOF mixedArray

L1: ;to print original array

mov eax, 0

mov eax, [esi]

call WriteInt

mov eax, 0

mov al, 44 ;ASCII value of ‘,’

call WriteChar

add esi, TYPE mixedArray

loop L1

mov edi, OFFSET array2

mov esi, OFFSET mixedArray

mov ecx, LENGTHOF mixedArray

L2: ;to move unsigned numbers to array2

mov eax, [esi]

cmp eax, 0

JL skip

mov [edi], eax

add edi, TYPE array2

skip:

add esi, TYPE mixedArray

loop L2

call crlf

call crlf

mov edx, OFFSET msg2

call WriteString

mov esi, OFFSET array2

mov ecx, LENGTHOF array2

L3: ;to print array2

mov eax, 0

mov eax, [esi]

cmp eax, 0

JE outofloop

call WriteInt

mov eax, 0

mov al, 44

call WriteChar

add esi, TYPE array2

loop L3

outofloop:

call crlf

exit

main endp

end main

