**Question 1:**

Write an assembly language code of the following C++ function. Consider all the integers of this function as unsigned DWORD in Assembly Language. Call the given C++ function in Assembly Language from main body using “CALL” operation. The output from function will be in accumulator register. Show the output from main body. (10)

int FunCPLUS(int var1, int var2, int var3) { int var4;

var4 = var1 + var2; var4 = var4 – var3;

if( var4 > var1 ) {

var4 = 10;

}

If (var4 < var2) {

var4 = 30;

}

return var4;

}

**Solution:**

include Irvine32.inc

.data

var1 DWORD 0

var2 DWORD 0

var3 DWORD 0

msg0 BYTE "Enter value 1: ", 0

msg1 BYTE "Enter value 2: ", 0

msg2 BYTE "Enter value 3: ", 0

msg3 BYTE "Final value: ", 0

.code

main PROC

mov eax, 0

call crlf

mov edx, OFFSET msg0

call WriteString

call ReadInt

mov var1, eax

call crlf

mov eax, 0

call crlf

mov edx, OFFSET msg1

call WriteString

call ReadInt

mov var2, eax

call crlf

mov eax, 0

call crlf

mov edx, OFFSET msg2

call WriteString

call ReadInt

mov var3, eax

call crlf

call crlf

call crlf

call FunCPLUS

mov edx, OFFSET msg3

call WriteString

call WriteDec

call crlf

exit

main ENDP

FunCPLUS PROC

;-----------------------------------------------------------------

;Function returns 10 or 30 depending on value of input arguments

;Receives: nothing

;Requires: three variables with values assigned to them

;Returns: resultant value in the EAX register

;-----------------------------------------------------------------

mov eax, 0

add eax, var1

add eax, var2

sub eax, var3

cmp eax, var1

JBE skip1

mov eax, 10

skip1:

cmp eax, var2

JAE skip2

mov eax, 30

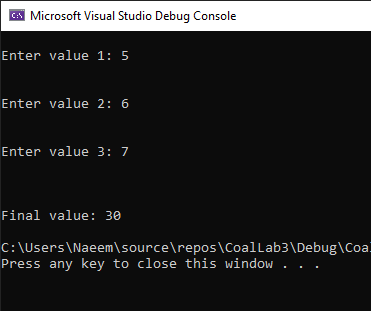
skip2:

ret

FunCPLUS ENDP

END main

Program output:



Entire screen:

