**COAL LAB 07 (LAB TASKS)**

**TASK # 01:**

**CODE:**

TITLE Q1 (test.asm)

INCLUDE Irvine32.inc

.data

array1 word 10h,20h,30h,40h,50h,60h,70h,80h,90h,100h

array2 word 10 dup(?)

.code

main PROC

mov ecx, lengthof array1 ; to set the counter

mov eax, 0 ; to clear out the register

mov esi, offset array1

l1: mov ax, [esi]

push ax

add esi, 2

loop l1

mov ecx, lengthof array1

mov esi, offset array2

l2: pop ax

mov [esi], ax

add esi, 2

loop l2

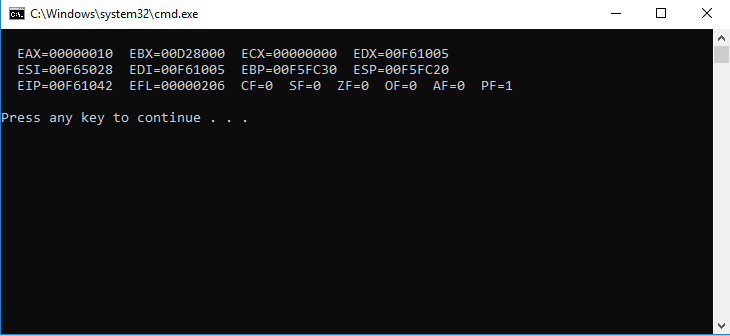
call DumpRegs

exit

main ENDP

End main

**OUTPUT:**



**TASK # 02:**

**CODE:**

TITLE Q2 (test.asm)

INCLUDE Irvine32.inc

.data

array1 word 'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H'

array2 word 8 dup(?)

.code

main PROC

mov ecx, lengthof array1

mov esi, offset array1

l1: mov ax, [esi]

push ax

add esi, 2

loop l1

mov ecx, lengthof array2

mov esi, offset array2

l2: pop ax

mov [esi], ax

add esi, 2

loop l2

mov ecx, lengthof array2

mov esi, offset array2

mov ebx, 0

; printing loop

l3: mov bx, [esi]

add esi, 2

loop l3

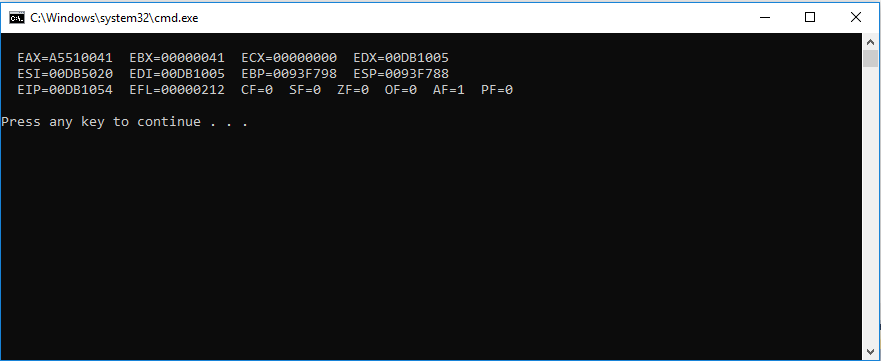
call DumpRegs

exit

main ENDP

End main

**OUTPUT:**



**TASK # 03:**

**CODE:**

TITLE Q3 (test.asm)

INCLUDE Irvine32.inc

.data

var1 word 01h

var2 word 02h

var3 word 03h

.code

main PROC

mov ax, var1

push ax

mov ax, var2

push ax

mov ax, var3

push ax

mov eax, 0

pop bx

mov ax, 1

mul bx

pop bx

mul bx

pop bx

mul bx

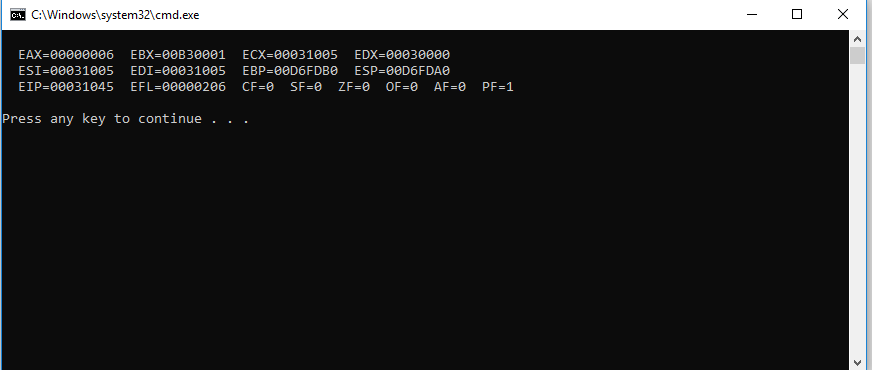
call DumpRegs

exit

main ENDP

End main

**OUTPUT:**



**TASK # 04:**

**CODE:**

TITLE Q4 (test.asm)

INCLUDE Irvine32.inc

.data

var1 word 02h

var2 word 03h

var3 word 04h

.code

main PROC

mov ax, var1

push ax

mov ax, var2

push ax

mov ax, var3

push ax

mov eax, 0

pop bx

add ax, bx

pop bx

add ax, bx

pop bx

add ax, bx

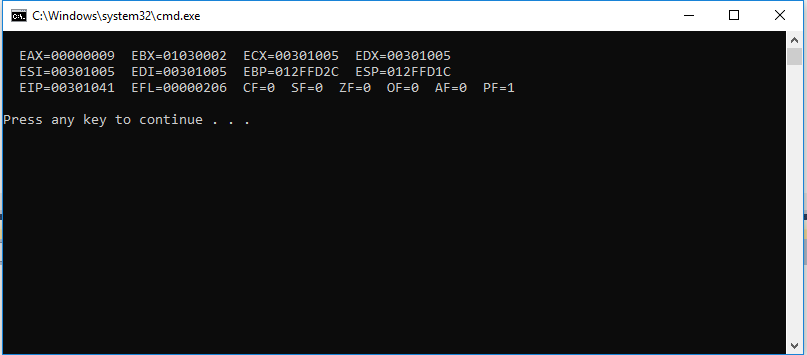
call DumpRegs

exit

main ENDP

End main

**OUTPUT:**



**TASK # 05:**

**CODE:**

TITLE Q5 (test.asm)

INCLUDE Irvine32.inc

.data

var1 dword 01h

var2 dword 02h

var3 dword 03h

result dword ?

.code

main PROC

call Mul1

call DumpRegs

exit

main ENDP

Mul1 PROC

mov eax, 0

call Mul2

mov eax, var1

mov ebx, result

mul ebx

mov result, eax

ret

Mul1 ENDP

Mul2 PROC

call Mul3

mov eax, var2

mov ebx, result

mul ebx

mov result ,eax

ret

Mul2 ENDP

Mul3 PROC

mov eax, 1

mov ebx, var3

mul ebx

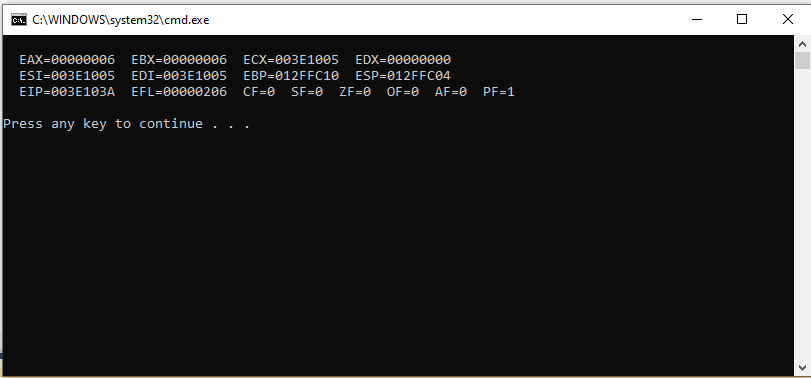
mov result, eax

ret

Mul3 ENDP

End main

**OUTPUT:**



**TASK # 06:**

**CODE:**

TITLE Q6 (test.asm)

INCLUDE Irvine32.inc

.data

array1 word 10h,20h,30h,40h,50h

array2 word 5 dup(?)

.code

main PROC

mov ecx, lengthof array1 ; to set the counter

mov eax, 0 ; to clear out the register

mov esi, offset array1

l1: mov ax, [esi] ; loop l1 to push

push ax

add esi, 2

loop l1

mov ebx, 0

mov bx, [esp+8] ; explicit parameters syntax.

mov [array2], bx

mov bx, [esp+6]

mov [array2+2], bx

mov bx, [esp+4]

mov [array2+4], bx

mov bx, [esp+2]

mov [array2+6], bx

mov bx, [esp]

mov [array2+8], bx

mov ecx, lengthof array2

mov esi, offset array2

mov eax, 0

l2: mov ax, [esi] ; loop to display contents in register

add esi, 2

loop l2

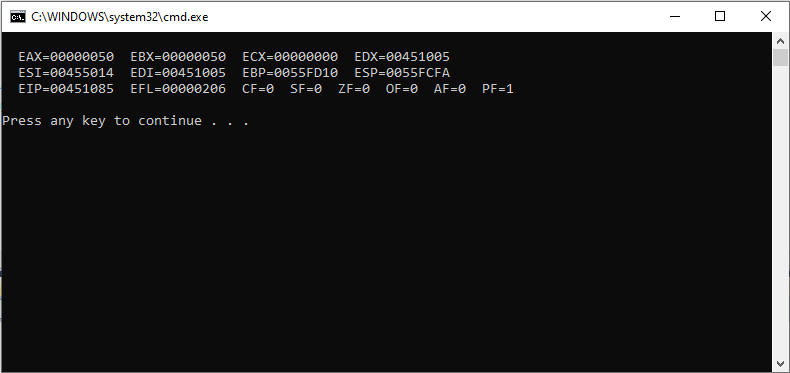
call DumpRegs

exit

main ENDP

End main

**OUTPUT:**



**TASK # 07:**

**CODE:**

TITLE Q7 (test.asm)

INCLUDE Irvine32.inc

.data

var1 dword 6h

.code

main PROC

call Factorial

call DumpRegs

exit

main ENDP

Factorial PROC

mov ecx, var1

mov eax, 1

l1:

mul ecx

loop l1

ret

Factorial ENDP

End main

**OUTPUT:**

