


# COAL Lab Tasks 6

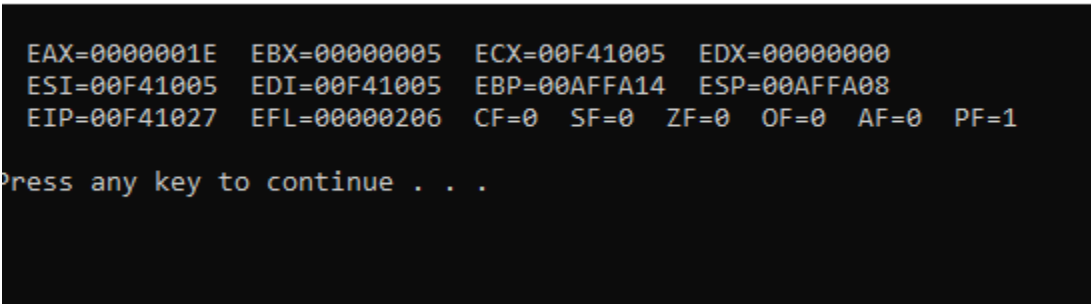
## Task 1:

### Code:

```
INCLUDE Irvine32.inc
.code
main PROC
    mov eax, 0
    push 5
    push 3
    push 2
    pop eax
    pop ebx
    mul ebx
    pop ebx
    mul ebx
    call DumpRegs
    exit
main ENDP
END main
```

### Screenshot:

 C:\WINDOWS\system32\cmd.exe



```
EAX=0000001E  EBX=00000005  ECX=00F41005  EDX=00000000
ESI=00F41005  EDI=00F41005  EBP=00AFFA14  ESP=00AFFA08
EIP=00F41027  EFL=00000206  CF=0  SF=0  ZF=0  OF=0  AF=0  PF=1

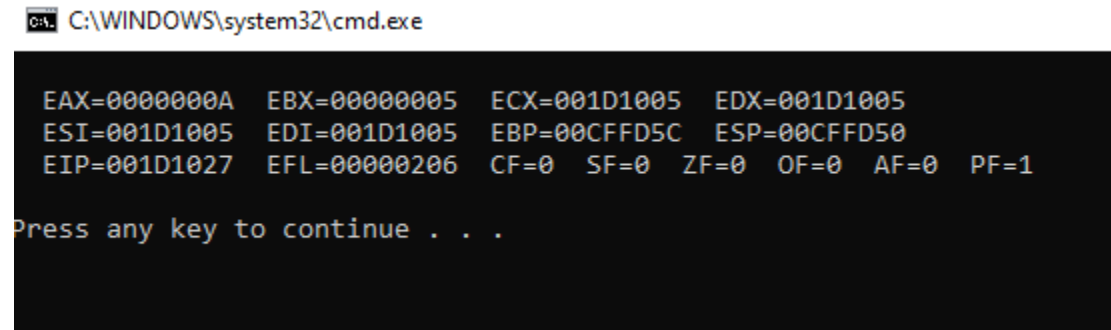
Press any key to continue . . .
```

## Task 2:

### Code:

```
INCLUDE Irvine32.inc
.code
main PROC
mov eax, 0
push 5
push 3
push 2
pop eax
pop ebx
add eax, ebx
pop ebx
add eax, ebx
call DumpRegs
exit
main ENDP
END main
```

### Screenshot:



The screenshot shows a Windows command prompt window with the title bar "C:\WINDOWS\system32\cmd.exe". The window displays the output of the Irvine32 library's DumpRegs function, which shows the current state of the x86 registers and flags. The output is as follows:

```
EAX=0000000A  EBX=00000005  ECX=001D1005  EDX=001D1005
ESI=001D1005  EDI=001D1005  EBP=00CFFD5C  ESP=00CFFD50
EIP=001D1027  EFL=00000206  CF=0  SF=0  ZF=0  OF=0  AF=0  PF=1
```

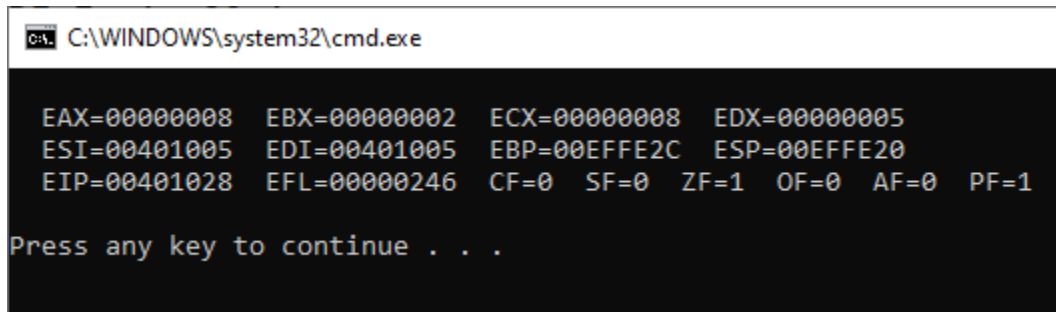
Below the register output, the prompt "Press any key to continue . . ." is displayed.

## Task 3:

### Code:

```
INCLUDE Irvine32.inc
.code
main PROC
mov eax, 0
push 5
push 8
push 2
push 1
pop eax           ;EAX=1
pop ebx           ;EBX=2
pop ecx           ;ECX=8
pop edx           ;EDX=5
mov eax, ecx      ;Movig largest value from ECX to EAX
call DumpRegs
exit
main ENDP
END main
```

### Screenshot:



```
C:\WINDOWS\system32\cmd.exe

EAX=00000008  EBX=00000002  ECX=00000008  EDX=00000005
ESI=00401005  EDI=00401005  EBP=00EFFF2C  ESP=00EFFF20
EIP=00401028  EFL=00000246  CF=0  SF=0  ZF=1  OF=0  AF=0  PF=1

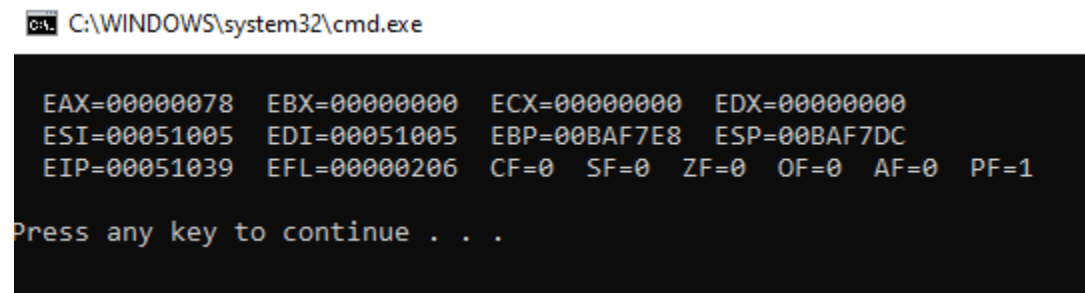
Press any key to continue . . .
```

## Task 4:

### Code:

```
INCLUDE Irvine32.inc
.data
fact DWORD 5h
res DWORD 1
.code
main PROC
mov eax, 0
mov ebx, 0
mov edx, 0
mov ecx, fact
F:
    mov eax, ecx
    mul res
    mov res, eax
Loop F
call DumpRegs
exit
main ENDP
END main
```

### Screenshot:



### Note:

Calculating factorial of 5.

Answer in EAX= 78, which is in Hexadecimal, in Decimal it will be 120, which is factorial of 5.