include irvine32.inc

;Question Number 1

        ;Construct a large integer (32 – Bit) from two Byte Variables (8 – Bit)

        ;and one Word Type variable (16 – Bit). Store your result into EAX and print it. Like if

        ;byte1 = 21h, byte2 = 43h & word1 = 8765h

        ;Then your output may be

        ;87654321h OR 43218765 OR 43876521

;##

;Question Number 2

    ;Reverse the given string:

    ;myStr BYTE "Step on no pets",

;Question Number 4

    ; find the factorial

;##

;Question Number 2

;Show the order of individual bytes in memory (lowest to highest) for the following double word variable using PTR Operato

.data

    print Label dword

    val_1 byte 21h

    val_2 byte 43h

    val_3 dword 8765h

```
;##################################################################################
###########################

        ;Now Question Number Two

        mystr byte "1223",0
        result byte sizeof  mystr Dup(0),0

;##################################################################################
###########################

        ;Question Number 3
         ;val1 DWORD 87654321

;##################################################################################
###########################

;Question Number 4
        ;find the factroial

        fact byte 1
        num byte 5

.code
main proc
;Answer 1
        ;xor eax ,eax
        ;mov eax, print
```

```
        ;call writedec
        ;call crlf
        ;call dumpregs
        ;exit


;####################################################
;Answer 2


        ;mov edx,offset mystr
        ;call writestring
        ;call crlf


        ;mov ecx,sizeof mystr
        ;call writedec
        ;mov esi,0
;L1:
        ;mov al,mystr[ecx-1]
        ;mov result+[esi],al
        ;call writechar
        ;call crlf
        ;inc esi
;loop L1
        ;mov edx, offset result
        ;call writestring
        ;call crlf


;######################################################
```

;Answer 3

```
;mov ax,word ptr val1

;mov al,byte ptr val1

;call writehex

;call crlf

;mov al,byte ptr [val1+1]

;call writehex

;call crlf

;mov al,byte ptr[val1+2]

;call writehex

;call crlf

;mov al,byte ptr[val1+3]

;call writehex
```

;Answer 4

```
        mov ecx,5

        xor eax,eax

        mov al,fact

l1:

        mul fact

        call writedec

        call crlf

        inc fact

loop l1
```

```
        exit


main endp

end main
```