

## Esercizio di programmazione

sino a 12 punti – è possibile consultare solamente l'Instruction Set Intel - tempo: 60 minuti

La distanza di Hamming tra due stringhe di ugual lunghezza è pari al numero di posizioni nelle quali i simboli corrispondenti sono diversi. In altri termini, la distanza di Hamming misura il numero di sostituzioni necessarie per convertire una stringa nell'altra, o il numero di modifiche necessarie per trasformare una stringa nell'altra. Ad esempio, si consideri la distanza di Hamming binaria tra i seguenti due interi:

11011101  
e 11001001

Il risultato in questo caso è 2.

Si scriva una procedura **hamming** in linguaggio Assembler 8086 che calcoli la distanza di Hamming binaria tra gli elementi di indice corrispondente di due vettori di *byte* di lunghezza DIM (dichiarato come costante), e salvi il risultato in un terzo vettore.

Esempio (valori in decimale e binario):

vet1	vet2	ris
56 (0011 1000)	1 (0000 0001)	4
12 (0000 1100)	0 (0000 0000)	2
98 (0110 0010)	245 (1111 0101)	5
129 (1000 0001)	129 (1000 0001)	0
58 (0011 1010)	12 (0000 1100)	4

La procedura riceve tramite *stack* l'indirizzo dei due vettori di dati e del vettore risultato. Di seguito un esempio di programma chiamante.

```
DIM EQU 5

.model small
.stack
.data
vet1 db 56, 12, 98, 129, 58
vet2 db 1, 0, 245, 129, 12
ris db DIM dup (?)

.code
.startup

LEA AX, vet1
PUSH AX
LEA AX, vet2
PUSH AX
LEA AX, ris
PUSH AX
CALL hamming
ADD SP, 6

.exit
```

## Soluzione proposta

hamming proc

PUSH BP  
MOV BP, SP

PUSH AX  
PUSH BX  
PUSH CX  
PUSH SI  
PUSH DI

MOV BX, [BP+4]  
MOV SI, [BP+6]  
MOV DI, [BP+8]

MOV CX, DIM  
cicloext: PUSH CX  
MOV AL, [DI]  
XOR AL, [SI]  
MOV CX, 8  
XOR AH, AH  
ciclo: SHL AL, 1  
ADC AH, 0  
LOOP ciclo  
MOV [BX], AH  
INC SI  
INC DI  
INC BX  
POP CX  
LOOP cicloext

POP DI  
POP SI  
POP CX  
POP BX  
POP AX  
POP BP

RET  
hamming endp