**Contor de locatii**

**Segment data**

**a db 17, -2, 0ffh, ‘xyz’,…**

**db ….**

**db….**

**;lga db $-a (mov [lga],….); ok //aritmetica de pointeri – scaderea a 2 pointeri = scalar (constanta numerica) - lga=variabila de memorie (mov [lga],…)**

**;lga dw $-$$ ; Corect numai DACA a este primul element definit in data segment !!!!**

**;lga EQU $-a ; ok ! insa mov [lga],… este syntax error !!! pt ca lga NU este variabila alocata…**

**;lga dw $-data ; corect in TASM/MASM, INCORECT in NASM sub 32 biti !!! syntax error – “Expression is not simple or relocatable”**

**;lga dw lga-a !!!!!!!!!**

**b EQU 27 ; b NU este un offset !!!!**

**c dd 12345678h**

**;lga dw b-a ; syntax error !!!!! b is NOT an address !!!**

**;lga dw c-a ; ok !!!!**

**lga dw $-a-4 ; ok !!!**

**lg dw $-a ; length (a) + 4 !!!**

 Daca nu se utilizeaza nicio directiva section in mod explicit, simbolul $$ se va evalua implicit la offset-ul inceputului de segment.

Elementul sintatic “:” se pune obligatoriu cand definim etichete de cod (ex: “start:”) insa nu trebuie pus daca definim o eticheta de date (ex: definirea de variabile “a db 17”)

Formatul al unei linii sursa nu este specific doar segmentului de cod, ci este general valabil pentru orice tip de segment din limbajul de asamblare (inclusiv segment de date)

**[*etichetă*[*:*]] [*prefixe*] [*mnemonică*] [*operanzi*] [;*comentariu*]**

Offset-ul oricarei etichete este o valoare constanta determinabila la momentul asamblarii. In orice limbaj de programare locul alocarii unei variabile (adresa sa) ramane fix; de aceea offset-urile variabilelor reprezinta valori constante determinabile la momentul asamblarii/compilarii

Adresa de segment este tot fixa dar este determinabila DOAR la momentul incarcarii (loading time)

Orice offset utilizat de sine statator in cadrul program va fi in cele din urma completat la o adresa FAR prin prefixare sa cu o valoare de segment. Aceasta valoare implicita va fi intoteauna unul dintre registri de segment CS, DS, SS

iar regulile pt efectuarea acestor asocieri implicite sunt:

* **CS** pentru etichete de cod destinaţie ale unor salturi (jmp, call, ret, jz etc);
* **SS** în adresări SIB ce foloseste EBP sau ESP drept bază (indiferent de index sau scală);
* **DS** pentru restul accesărilor de date