Location Counter

```
Segment data
  a db 17, -2, 0ffh, ‘xyz’,…
    db ….
    db….
```

;lga db $-a (mov [lga],….); ok //pointer arithmetic – subtracting 2 pointers = scalar value (numerical constant) – lga = memory variable (mov [lga],…)

;lga dw $-$$; the same correct length, but ONLY IF a is the first element allocated in the data segment !!!!

? ;lga EQU $-a ; ok !   but mov [lga],… will issue a syntax error !!! because lga is NOT an allocated variable… so it doesn't have a memory address to be deref.

;lga dw $-data ; correct in TASM/MASM, INCORRECT in NASM on 32 bits !!! syntax error – “Expression is not simple or relocatable”

;lga dw lga-a   !!!! ok !

b EQU 27 ; b is NOT an 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 !!!

If no section directive is explicitly used, the symbol $$ will be implicitly evaluated to the offset of the beginning of the current segment.

“:” is mandatory when we define code labels (ex: “start:”) but must not be present when we define a data label (ex: a variable definition “a db 17”)