byte db = 8 bits → [0,255] unsigned
→ [-128, 127] signed

word dw = 16 bits → $[0, 2^{16}-1]$ uns
$[-2^{15}, 2^{15}-1]$ signed

Dword dd = 32 bits

Qword dq = 64 bits

```
mov EAX,0
mov ECX,0
```
} cleaning registers

mov is like copy

mul
- multiplication
- specific 1 operand
- celălalt e implicit
- rezultatul e trimis la registrul x2

AL, BL, CL

mul CL

mul <OP> → mereu pune în EAX

imul – for signed variables

MUL
→ register
→ variable

– we multiply nr of the same type

→ Byte * Byte

MUL <OP> $AX = AL * \langle OP \rangle$

-> WORD*WORD

$AX * \langle OP \rangle = \underbrace{DX:AX}$ lower half in AX
higher half in DX

ex: mov AX, 2^{15}

mov BX, 2^{15}

mul BX

$\underbrace{DX}_{1\ldots} : \underbrace{AX}_{00\ldots0} = 2^{30}$

-> DWORD*DWORD

$EAX * \langle OP \rangle = EDX:EAX$

Division

DIV<OP>

-> $\overset{WORD}{AX} / \overset{BYTE}{\langle OP \rangle} = \overset{BYTE}{\underbrace{AL}_{cât}, \underbrace{AH}_{rest}}$

-> $DX:AX / \langle OP \rangle = AX, DX$

-> $\overset{QWORD}{EDX:EAX} / \overset{DWORD}{\langle OP \rangle} = EAX, EDX$