

# Curs 7

## Scopul principal al asamblorului

De a genera în mod corect octeții corespunzători directivelor și instrucțiunilor întâlnite

Directivele dirijează asamblorul, instrucțiunile dirijează procesorul

Segment data

a db 17, -2, affh, 'xyz', ...

db ...

dw ...

; lga dw \$ - a ; permite mov [lga]

; lga equ \$ - a ; NU permite [lga]

; lga db \$ - \$\$ ; NUMAI dacă a este primul element definit în DATA

lga dw lga - a

mov eax, [v] ; mov eax, dword ptr [DS: 405000]

mov eax, [ebx] ; DS

mov eax, [ebp] ; SS

mov eax, [ebp \* 2] ; mov eax, dword ptr: [SS: ebp + ebp]

mov eax, [ebp \* 3]; SS

mov eax, [ebp \* 4]; DS

mov eax, [ebp + esp]; ... SS

mov eax, [esp + ebx]; ...SS

mov eax, [ebx + esp \* 2]

mov eax, [ebx + ebp \* 2] - DS

mov eax, [ebx + ebp] - DS - Contează cum le-ai pus ca și baza

mov eax, [ebp + ebx] - SS - Contează cum le-ai pus ca și baza

mov eax, [ebx \* 2 + ebp]; SS

mov eax, [ebx \* 1 + ebp]; SS

mov eax, [ebp \* 1 + ebx]; DS

mov eax, [ebp \* 1 + ebx \* 1]; DS

```
mov eax, [ebp * 1 + ebx * 2]; SS
```

```
mov eax, [ebx * 1 + ebp * 1]; SS
```

### Subiect de examen

Prezentați în scris 20 de metode cu 0 a unui registru

```
mov eax, 0
```

```
and eax, 00000000h
```

```
xor eax, eax
```

Q: Precizați care sunt operațiile la nivel de bit care lasă neschimbată valoarea unor biți

R: Toate 3 prezintă un caz de utilizare