

III. a) 1. `lea eax, [6+esp]`; se adună 6 la ESP și se pune rezultatul în EAX

2. `mov eax, 6+esp`; syntax error, nu respectă formula de calcul a offset-ului

3. `movsx ax, [6+esp]`; se pune în AX cu sign extended un byte de la adresa `esp+6`

4. `mov ebp, [6+ebp*2]`; se pune în EBP dwordul de la adresa `6+ebp*2`

5. `mov [6+ebp*2], 12`; syntax error, trebuie specificată dimensiunea (byte / word / dword)

6. `mov ebp, [ebx+esp]`; ESP nu poate fi index, deci la ESP ca bază și EBX index; pune în EBP dwordul de la adresa `EBX+ESP`

7. `movsx [6+esp], eax`; la adresa `esp+6` se pune valoarea lui EAX

8. `mov [6+esp*2], eax`; ESP nu poate fi index, deci syntax error, invalid effective address

9. `mov [6+ebp*2], [6+esp]`; syntax error, nu pot fi ambii operanzi expliciti din memorie, deoarece se poate calcula o singură adresă în BIU

10. `movzx eax, [6+ebp*2]`; syntax error, trebuie specificată dimensiunea celui de-al doilea operand (byte / word)

- b)
- push edx ; pune EDX pe stivă
 - push eax ; pune EAX pe stivă
 - pop edx ; pune în EDX vârful stivei, adică EAX
 - echivalent cu mov edx, eax

EDX: [...] [0] [AL]

• xor dh, dh ; pune 0 în DH

• shl edx, 16 ; shiftază la stânga

EDX: [00] [AL] [00] [00]
 cu 16 biti
 ; mută wordul low
 ; în locul wordului high,
 ; iar în wordul low rămâne 0

* valoare random *

EDX = 04210307h =

= 0000 0100 0010 0001 0000 0000 0000 0111b
 DH DL
 → 0000 0100 0010 0001 0000 0000 0000 0111b
 DH DL
 0000 0000 0000 0111 0000 0000 0000 0000

• cfc ; clear carry flag, CF=0

• rcr edx, 16 ; rotește cu 16 biti la dreapta bitii din EDX, iar bitii

EDX [00] [00] [00] [AL] iesiti la dreapta (din CF) sunt pusi la stânga.

; cum primii 16 biti sunt 0, wordul high din EDX vine 0.

• add edx, ebx ; echivalent cu EBX+AL, rezultatul este în EDX

• push edx ; pune EBX+AL pe stivă

• pop esi ; pune în ESI vârful stivei, adică EBX+AL

• lodsb ; în AL se încarcă byte-ul de la adresa EBX+AL

• pop edx ; restorază valoarea din EDX

Echivalent cu XLAT

• exercițiile sînt aceleași ca cele din 18 februarie 2019