POSTESCU RAZVAN-IOAN A2 713/1

```
bits 32
global start
extern exit
import exit msvcrt.dll
segment data use32 class=data
        sir1 dw -3,-4,10
        l1 equ ($-sir1)/2
        sir2 dw 7,8,89
        l2 equ ($-sir2)/2
        media1 resw l1
        media2 resw l2
        cinci db 5
segment code use32 class=code
  start:
    mov ECX, l1 ; mutam lungimea sir.1 in ECX pt loop
    mov ESI ; counter =0
    mov BL ; in BL pun nr elementelor div. cu 5
    ~~mov DL, Emu~~
    loop_sir1:
        mov AX, [sir1+ESI] ; pune in AX al ESI-lea
                                   elem din sir1
        div byte [cinci] ; efect. impartirea cu 5 cu rest
                              in AH
        cmp AH, 0 ; face comparatie rest cu 0
        JZ sedecazero ; sare atunci cand zero
                          flag se face 1 din comparatie
          JMP notzero ; sare in cat counter
      sedecazero:
        add DX, [sir1+ESI] ; adunam elementul
                                          la DX
        inc BL ; incrementam BL
  notzero :  ; in cazul notzero continuam executia
  inc ESI

mov  AX, DX;
div BL ; media aritmetica se afla pe ~~AH~~ AL acum
 mov media1, AL ; salvam in media1 rezultatul
mov ECX, l2 ; mutam len. sir2 in ECX

mov ESI, 0; ~~B~~ mov BL, 0
 mov DX, 0 ; in DX punem media
 loop_mi2 :

    mov  AX, [sir2+ESI] ; punem sir2 in AX
    test  ESI, 1 ; seteaza zeroflag=1 cand e par
    JZ  sarepara ; sare cand poz. e para
      JMP sare_cand_impar
     sarepara:
        ~~mov [ESI]~~ mov DX, [sir2+ESI] ; punem in
      DX elem. ce se afla pe
     sarecandimpar:
      inc ESI ; incrementam ESI
       inc BL ;
 loop loop_mi2 ;

  mov AX, DX
  div BL ; in AH avem media2
mov media2, AL ; salvam media2 in rezultat


push dword 0
call [exit]
```