Tema laborator 3

Pop Mihai-Daniel, Grupa 215/2

Adunari, scaderi – interpretare cu semn

**a - byte, b - word, c - double word, d - qword**

1. Ex. 27: (a+c)-(d+b)

; a - byte, b - word, c - double word, d - qword

; Ex. 27: (a+c)-(d+b)

; ex.1 : a=125, b=2, c=15, d=200; Rezultat: (125+15)-(2+200) = 140-202 = -62

bits 32

global start

extern exit ; indicam asamblorului ca exit exista, chiar daca noi nu o vom defini

import exit msvcrt.dll; exit este o functie care incheie procesul, este definita in msvcrt.dll

; msvcrt.dll contine exit, printf si toate celelalte functii C-runtime importante

segment data use32 class=data ; segmentul de date in care se vor defini variabilele

a db 125

b dw 2

c dd 15

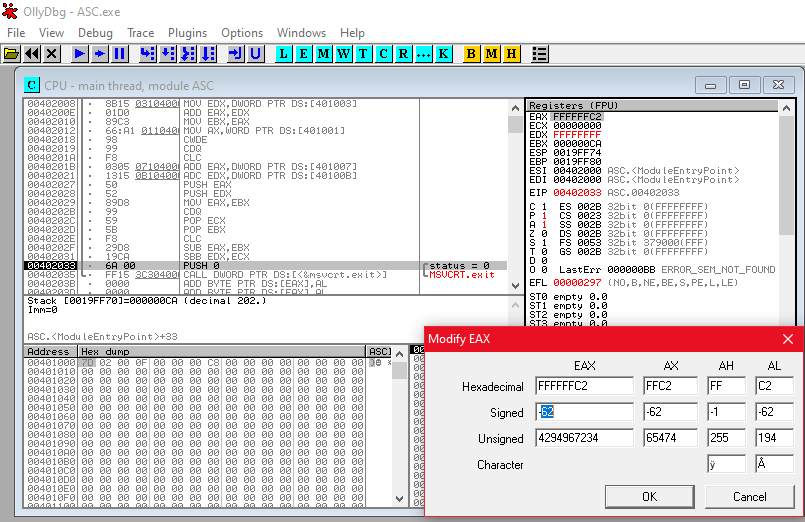
d dq 200

segment code use32 class=code ; segmentul de cod

start:

;pentru a calcula a+c, convertim a de la byte la doubleword pentru a-l putea aduna la doubleword-ul c

mov al, [a] ;al = a = 125

cbw ;conversie cu semn de la al la ax

cwde ;conversie cu semn de la ax la eax

;eax = a = 125

mov edx,[c] ;edx = c = 15

add eax,edx ;adunare eax cu edx

;eax = eax + edx = 15+125 = 140

mov ebx,eax ;ebx = eax = 140

mov ax, [b] ;

cwde ;conversie cu semn de la ax la eax

cdq ;conversie cu semn de la eax la edx:eax

;edx,eax = b = 2

clc ;Carry Flag = 0

add eax, dword [d]

adc edx, dword [d+4] ;edx:eax = d + b = 2+200 = 202

push eax

push edx

;am pus in stiva valoarea rezultatului (d+b)

mov eax, ebx ;

cdq ;conversie cu semn de la eax la edx:eax

;edx:eax = 140

pop ecx

pop ebx

;am scos din stiva valoarea rezultatului (d+b) = 202

clc

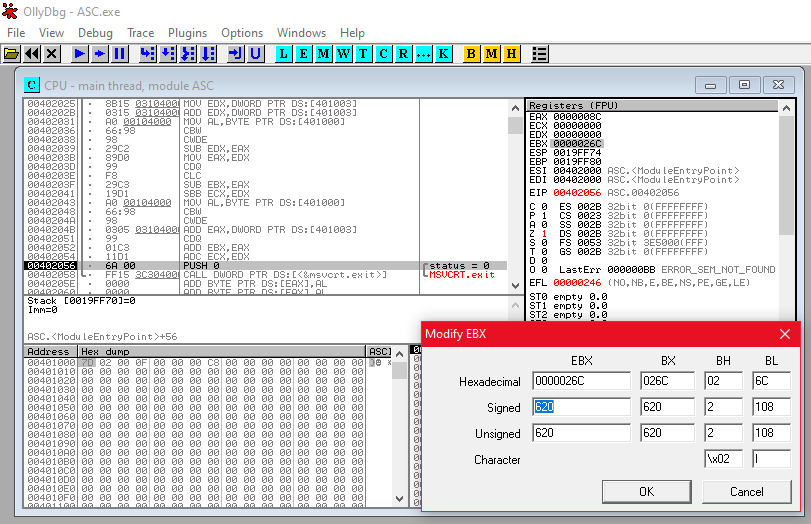
sub eax,ebx

sbb edx,ecx

;(a+c)-(d+b) = 140-202 = -62

push dword 0 ;se pune pe stiva codul de retur al functiei exit

call [exit] ;apelul functiei sistem exit pentru terminarea executiei programului

1. Ex. 27: (d+d-c)-(c+c-a)+(c+a)

; Adunari si scaderi in interpretarea cu semn

; a - byte, b - word, c - double word, d - qword

; Ex. 27: (d+d-c)-(c+c-a)+(c+a)

; ex.1 : a=125, b=2, c=15, d=200;

; Rezultat: (200+200-15)-(15+15-125)+(15+125) = 385-(-95)+140 = 620

bits 32

global start

extern exit

import exit msvcrt.dll

segment data use32 class=data

a db 125

b dw 2

c dd 15

d dq 200

segment code use32 class=code ; segmentul de cod

start:

mov eax, dword [d]

mov edx, dword [d+4] ;edx:eax = d = 200

add eax, dword [d]

adc edx, dword [d+4] ;edx:eax = d+d = 200+200 = 400

mov ebx, eax

mov ecx, edx ;mutam rezultatul calculului d+d in perechea de registrii ecx:ebx

mov eax, [c]

cdq ;convertim variabila c din dword in qword

sub ebx, eax

sbb ecx, edx ;ecx:ebx = d+d-c = 400-15 = 385

mov edx, [c] ;edx = c = 15

add edx, [c] ;edx = edx+c = c+c = 15+15 = 30

mov al, [a]

cbw

cwde ;eax = a = 125

sub edx,eax ;ebx = edx-eax = c+c-a = 30-125 = -95

mov eax, edx ;eax = edx = -95

cdq ;convertim rezultatul din a doua paranteza din dword in qword ;edx:eax = -95

clc

sub ebx, eax

sbb ecx, edx ;ecx:ebx = (d+d-c)-(c+c-a) = 385-(-95) = 385+95 = 480

mov al, [a]

cbw

cwde ;eax = a = 125

add eax, [c] ;eax = eax+c = (a+c) = 125+15 = 140

cdq

add ebx, eax

adc ecx, edx ;ecx:ebx = (d+d-c)-(c+c-a)+(a+c) = 480+140 = 620

push dword 0 ;se pune pe stiva codul de retur al functiei exit

call [exit] ;apelul functiei sistem exit pentru terminarea executiei programului

Inmultiri, impartiri – interpretare cu semn

**a,b - byte, c - word, e - double word, x – qword**

1. Ex. 27: (100+a+b\*c)/(a-100)+e+x/a

; Inmultiri si impartiri in interpretarea cu semn

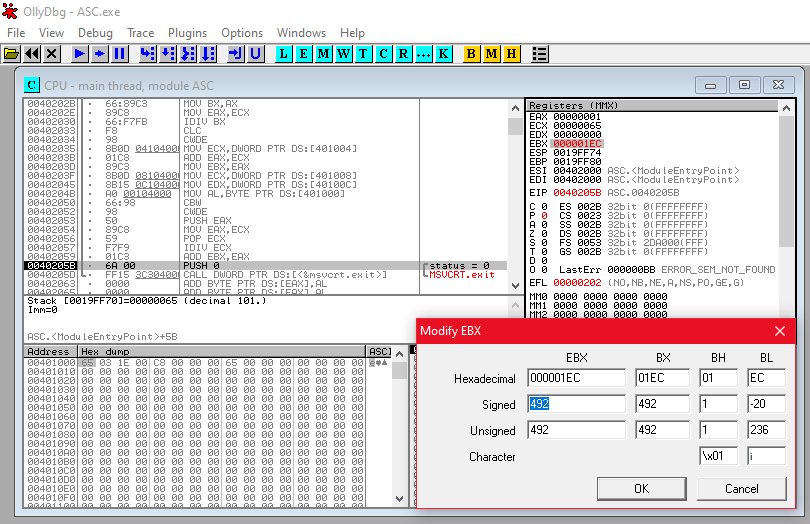
; a,b - byte; c - word; e - doubleword; x - qword

; Ex. 27: (100+a+b\*c)/(a-100)+e+x/a

; ex.1 : a=101, b=3, c=30, e=200, x=101;

; Rezultat: (100+101+3\*30)/(101-100)+200+101/101 = 291/1+200+1 = 492

bits 32

global start

extern exit

import exit msvcrt.dll

segment data use32 class=data

a db 101

b db 3

c dw 30

e dd 200

x dq 101

segment code use32 class=code ; segmentul de cod

start:

mov al, [a]

cbw

cwde ;eax = a = 101

add eax, 100 ;eax = eax+100 = 201

mov ecx, eax ;ecx = eax = 201

mov al, [b]

cbw ;ax = b = 3

mov bx, [c] ;bx = c = 30

imul bx ;eax = ax\*bx = b\*c = 3\*30 = 90

add ecx, eax ;ecx = ecx+eax = (100+a+b\*c) = 201+90 = 291

mov al, [a]

cbw ;ax = a = 101

sub ax, 100 ;ax = ax-100 = a-100 = 101-100 = 1

mov bx, ax ;bx = ax = 1

mov eax, ecx ;eax = ecx = 291

idiv bx ;ax = eax/bx = (100+a+b\*c)/(a-100) = 291/1 = 291

clc

cwde ;ax = eax = 291

mov ecx, [e] ;ecx = e = 200

add eax, ecx ;eax = eax+ecx = 291+200 = 491

mov ebx, eax ;ebx = 204

mov ecx, dword [x]

mov edx, dword [x+4] ;edx:ecx = x = 101

mov al, [a]

cbw

cwde ;eax = a = 101

push eax

mov eax, ecx ;eax = ecx

pop ecx ;ecx = a = 101

idiv ecx ;eax = edx:eax/ecx = x/a = 101/101 = 1

add ebx, eax ;ebx = ebx+eax = (100+a+b\*c)/(a-100)+e+x/a = 491+1 = 492

push dword 0

call [exit]

**a - byte, b - word, c - double word, d - qword**

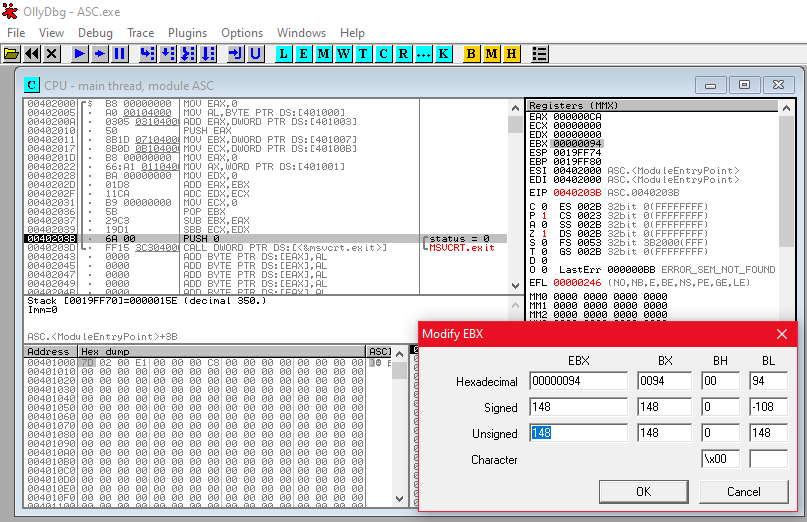
1. Ex. 27: (a+c)-(d+b) interpretare fara semn

; Adunari si scaderi in interpretare fara semn

; ; a - byte, b - word, c - double word, d - qword

; Ex. 27: (a+c)-(d+b)

; ex.1 : a=125, b=2, c=225, d=200; Rezultat:(125+225)-(200+2) = 350-202 = 148

bits 32

global start

extern exit

import exit msvcrt.dll

segment data use32 class=data

a db 125

b dw 2

c dd 225

d dq 200

segment code use32 class=code ; segmentul de cod

start:

mov eax, 0 ;eax = 0

mov al, [a] ;eax = a = 125

add eax, [c] ;eax = eax+c = (a+c) = 125+225 = 350

push eax ;punem in stiva rezultatul, eliberand registrul

mov ebx, dword [d]

mov ecx, dword [d+4] ;ecx:ebx = d = 200

mov eax, 0 ;eax = 0

mov ax, [b] ;eax = b = 2

mov edx, 0 ;edx = 0

;edx:eax = b = 2

add eax, ebx

adc edx, ecx

;edx:eax = d+b = 200+2 = 202

mov ecx, 0 ;ecx = 0

pop ebx ;ebx = 350

sub ebx, eax

sbb ecx, edx

;ecx:ebx = (a+c)-(d+b) = 350-202 = 148

push dword 0

call [exit]