****

**MINISTERUL EDUCAȚIEI, CULTURII ȘI CERCETĂRII**

**AL REPUBLICII MOLDOVA**

**Universitatea Tehnică a Moldovei**

**Facultatea Calculatoare, Informatică şi Microelectronică**

**Raport**

**Lucrarea de laborator №1**

Tema: Elaborare programelor în limbajul de asamblare MASM în Visual Studio

**Realizat:** Bunescu Gabriel, TI-207

**Profesor:** asist.univ.mag.Secrieru Adrian

Chișinău 2022

**Scopul lucrării:**

Lucrarea urmărește familiarizarea studenților cu regișorii microprocesoarelor pe 16, 32 și 64 de biți, cu mediul de dezvoltare a programelor Visual Studio. Se prezintă setările mediului Visual Studio necesare pentru a elabora, rula și depăna aplicațiile elaborate în limbajul de asamblare MASM utilizând registre pe 16, 32 și 64 de biți.

**Mersul lucrării:**

* **Programele pe 32 de biți**

Programele pe 32 de biți se execută în modul consolă fără utilizarea de programe suplimentare, aveți nevoie doar de o caracteristică încorporată în Visual Studio, "Rularea fără depanare." Când utilizați depanarea sau depanarea pas cu pas, trebuie să traduceți codul într-un formular dezasamblat.

Program: 32 biti

INCLUDE Irvine32.inc

.data

Promt DB 'Doresti sa devii programator?(da/nu)-[y/n]',0

Dad DB 13,10,'Vei deveni!',13,10,0

Nud DB 13,10,'Vei deveni filosof!',13,10,0

.code

main PROC

mov edx,OFFSET Promt

call WriteString

call ReadChar

cmp al,'y'

jz IsDad

cmp al,'n'

jz IsNud

IsDad: mov edx,OFFSET Dad

call WriteString

jmp ex

IsNud: mov edx,OFFSET Nud

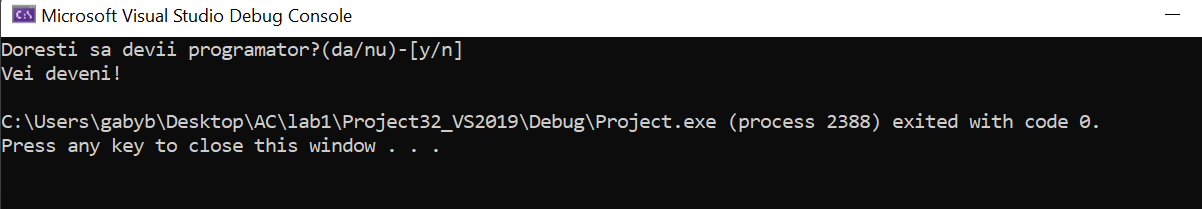
call WriteString

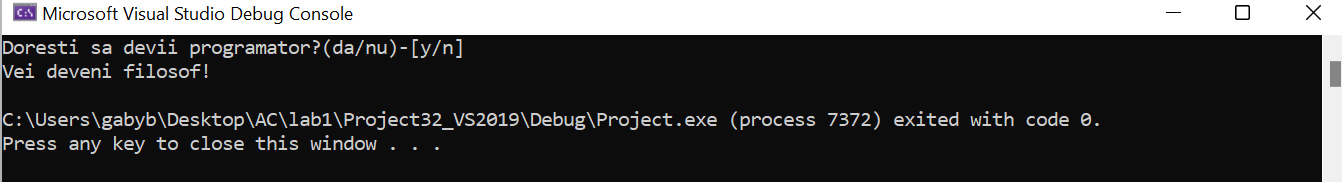
ex:

exit

main ENDP

END main





Program: 64 biti

; AddTwoSum\_64.asm - Chapter 3 example.

ExitProcess proto

.data

sum qword 0

.code

main proc

mov rax,5

add rax,6

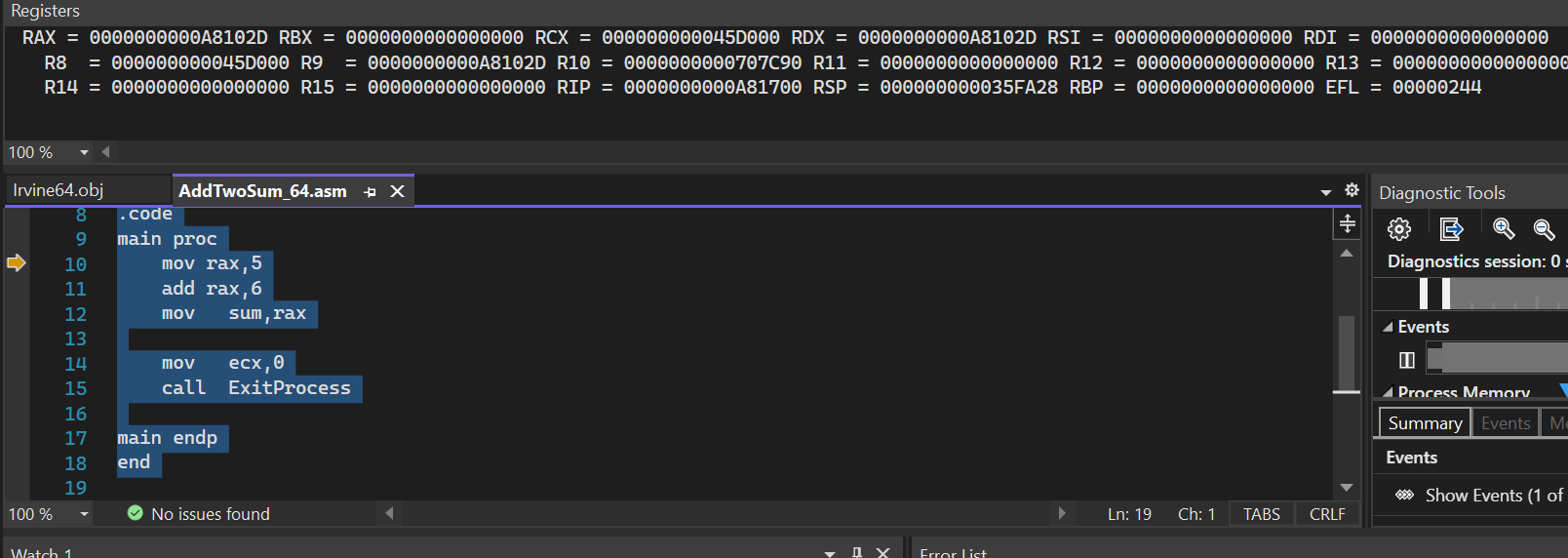
mov sum,rax

mov ecx,0

call ExitProcess

main endp

end



* **Programele pe 16 de biți**

Programele pe 16 biți nu sunt acceptate de procesoare pe 32 și 64 de biți, astfel încât nu se pot executa fără programe de asistență. Pentru a executa programe pe 16 biți, trebuie să utilizați un emulator (cum ar fi DosBox 0.74 instalat pe acest computer).

Exemplu:

TITLE Add and Subtract (16-bit.asm)

; This program adds and subtracts 32-bit integers.

INCLUDE Irvine16.inc

.code

main PROC

mov ax,@data

mov ds,ax

mov eax,10000h ; EAX = 10000h

add eax,40000h ; EAX = 50000h

sub eax,20000h ; EAX = 30000h

call DumpRegs

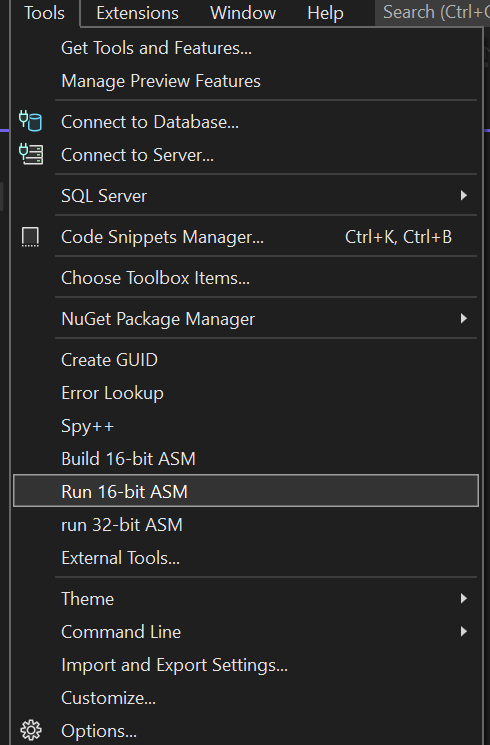
exit

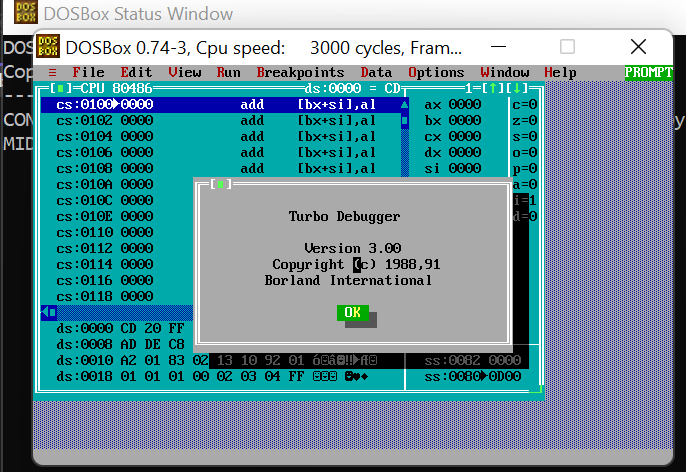
main ENDP

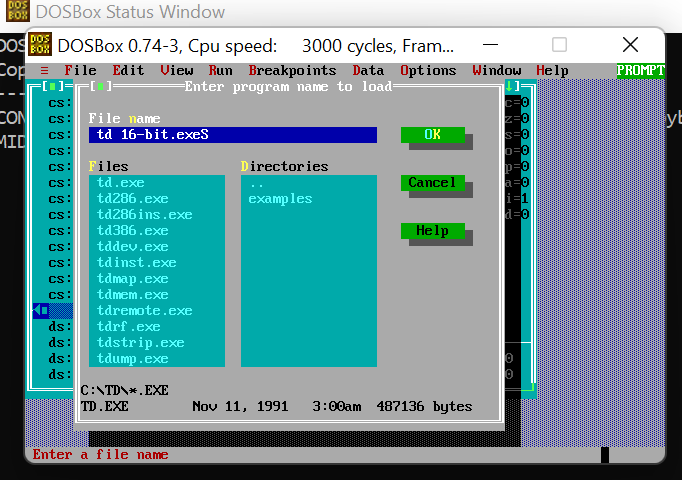
END main

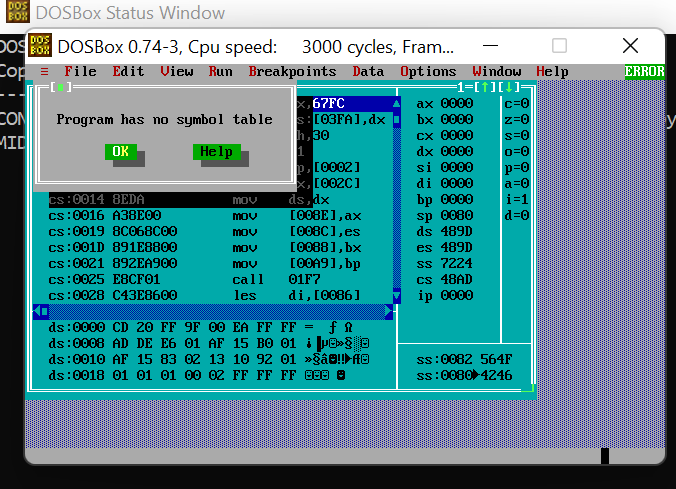
Acest program efectuează operații de adăugare și scădere pe 32 de biți. Acesta utilizează proceduri de la Irvine16.inc, concepute special pentru programe pe 16 biți.

Executarea programului utilizând DosBox 0.74-3:









**Concluzie:**

În cursul acestei lucrări de laborator, am studiat metodele de scriere și lansare a programelor pe 16, 32 și 64 de biți scrise în limbajul de programare al MASM în Visual Studio. Procesoarele computerelor de astăzi acceptă software pe 32 de biți, în timp ce software-ul pe 16 biți necesită software suplimentar, ar fi emulatorul DosBox 0.74-3. Visual Studio a fost, de asemenea, configurat special pentru fișierele .asm și tehnica de rutină și pas-cu-pas depanare a fost studiat.