

Ministerul Educației al Republicii Moldova
Universitatea Tehnică a Moldovei
Facultatea Calculatoare, Informatică și Microelectronică
Departamentul Ingineria Software și Automatică

Raport

Lucrarea de laborator nr.4

Disciplina: Arhitectura Calculatoarelor

A efectuat:

st. gr.TI-174 Iepuraș Daniel

A verificat:

conf.univ. V.Colesnic

Tema: „SUBPROGRAME ȘI MACROINSTRUCȚIUNI”

Scopul Lucrării:

Lucrarea urmărește deprinderea studenților cu proiectarea și implementarea programelor cu subprograme și macroinstrucțiuni, cu ramificații și bucle în limbaj de asamblare. Se prezintă câteva exemple tipice de astfel de programe, incluzând câteva metode elementare de sortare și căutare. De asemenea se prezintă câteva implementări de operații aritmetice care necesită utilizarea unor structuri de control de tip ramificare și buclare. Se vor utiliza instrucțiunile de comparare, salt și buclare. Se prezintă de asemenea și câteva exemple de utilizare a instrucțiunilor logice.

Sarcina Lucrării:

Varianta 7

Elaborați un program de înmulțire a două numere întregi, în hexa, reprezentate pe 64 de biți, **qp1** și **qp2**. Afișați rezultatul în hexa.

Codul sursa al programului

```
ExitProcess proto
WriteString proto
WriteHex64 proto
Crlf proto

.data
qp1 qword 4h
qp2 qword 3h
var1 byte "X=",0
var2 byte "Y=",0
var3 byte "Rezultatul X*Y=",0

.code
main proc

mov rdx,offset var1
call WriteString
mov rax,qp1
call WriteHex64
call Crlf

mov rdx,offset var2
call WriteString
mov rax,qp2
call WriteHex64
call Crlf
```

```

mov rdx,offset var3
call WriteString
    mov rax, qp1
    imul qp2
    call WriteHex64
    call Crlf

mov ecx,0
call ExitProcess

main endp
end

```

Rezultatul programului:

```

X=0000000000000004
Y=0000000000000003
Rezultatul X*Y=000000000000000C

```

Listingul programului:

Microsoft (R) Macro Assembler (x64) Version 14.16.27027.1
C:\Irvine\ch03\64 bit\AddTwoSum_64.asm

04/29/19 11:47:47
Page 1 - 1

```

ExitProcess proto
WriteString proto
WriteHex64 proto
Crlf proto

00000000 .data
00000000 qp1 qword 4h
00000008 qp2 qword 3h
00000010 58 3D 00 var1 byte "X=",0
00000013 59 3D 00 var2 byte "Y=",0
00000016 52 65 7A 75 6C var3 byte "Rezultatul X*Y=",0
00000016 74 61 74 75 6C
00000016 20 58 2A 59 3D
00000016 00

00000000 .code
00000000 main proc

00000000 48/ BA mov rdx,offset var1
00000010 0000000000000010 R
0000000A E8 00000000 E call WriteString
0000000F 48/ 8B 05 mov rax,qp1
0000000F 00000000 R
00000016 E8 00000000 E call WriteHex64

```

```

0000001B  E8 00000000 E    call Crlf

00000020  48/ BA          mov rdx,offset var2
0000000000000013 R
0000002A  E8 00000000 E    call WriteString
0000002F  48/ 8B 05        mov rax,qp2
0000000008 R
00000036  E8 00000000 E    call WriteHex64
0000003B  E8 00000000 E    call Crlf

00000040  48/ BA          mov rdx,offset var3
0000000000000016 R
0000004A  E8 00000000 E    call WriteString
0000004F  48/ 8B 05        mov rax, qp1
0000000000 R
00000056  48/ F7 2D        imul qp2
0000000008 R
0000005D  E8 00000000 E    call WriteHex64
00000062  E8 00000000 E    call Crlf

00000067  B9 00000000      mov ecx,0
0000006C  E8 00000000 E    call ExitProcess

00000071                          main endp
                                End

```

Concluzii:

În urma realizării lucrării de laborator nr.4 la tema: “*Subprograme și macroinstrucțiuni*”, am însușit mai bine proiectarea și implementarea programelor cu subprograme și macroinstrucțiuni, cu ramificații și bucle în limbaj de asamblare. Am prezentat utilizarea unor structuri de control de tip ramificare și buclare. La fel, am utilizat instrucțiunile de comparare, salt și buclare, prezentând și câteva exemple de utilizare a instrucțiunilor logice. În concluzie, conform variantei am elaborat un program care presupune înmulțirea a două numere întregi, în hexa, reprezentate pe 64 de biți, **qp1** și **qp2**. Rezultatul înmulțirii de asemenea l-am reprezentat în hexa.