Laborator 11 – ASC IR
ASM+ASM

Pas 1: Creare director care sa contina: ALINK.exe si NASM.exe (ambele executatbile se gasesc in directorul nasm din asm tools)
Codurile sursa ale problemelor vor fi salvate in acelasi director cu cele 2 executabile alink si nasm.
Pas 2: rezolvare probleme:
mainsuma.asm

```
bits 32
global start
import printf msvcrt.dll
import exit msvcrt.dll
import scanf msvcrt.dll
extern printf, exit, scanf

extern calculsuma ; declarare eticheta de calclul din modulul secundar
segment data use32
  format_afisare db "suma=%d", 10, 13, 0
  mesaja db "introduceti a=",10, 13, 0
  mesajb db "introduceti b=",10, 13, 0
  format db "%d",0
    a resd 1
    b resd 1
segment code use32 public code
start:
      ;in main vom citi numerele a, b
            ;printare mesaj de citire pe ecran
      push dword mesaja
      call [printf]
      add esp,4
      ;citire a
      push dword a
      push dword format
      call [scanf]
      add esp,4*2
      ;printare mesaj de citire pe ecran
      push dword mesajb
      call [printf]
      add esp,4
      ;citirea lui b de la tastatura
      push dword b
      push dword format
      call [scanf]
      add esp,4*2

      push dword [a] ;salvam pe stiva
                  ;numerele citite pentru a fi accesate
```

```
                ;din modulul secundar
        push dword [b]

        call calculsuma ; apelare functie de
                ;calcul suma din modulul secundar


;afisare
            push ebx
            push format_afisare
            call [printf]
            add esp, 2*4
     ;   popa
       push dword 0
       call [exit]
```

secondsuma.asm

```
;cod pentru sumafunctie.asm
bits 32
segment code use32 public code
global calculsuma  ; eticheta
calculsuma:

    mov eax, [esp + 4] ;accesam primul param de pe stiva

    mov ebx, [esp + 8] ;accesam al doilea param de pe stiva

    add ebx, eax ; calcul

ret 4*2 ; in acest caz 8 reprezinta
    ;numarul de octeti ce trebuie eliberati de pe stiva
    ;(parametrii pasati procedurii - adica cei 2 pusi pe stiva*4)
```

Pas 3: pozitionare in director (in linia de comanda): start -> cmd

Pas 4: comenzii pentru a transforma asm in obj si apoi obj+obj -> exe

```
D:\_2018 didactic\ASM_tools\asm_tools_adriana 2018\npp\multimmm\exemplu>nasm -f obj mainsuma.asm

D:\_2018 didactic\ASM_tools\asm_tools_adriana 2018\npp\multimmm\exemplu>nasm -f obj secondsuma.asm

D:\_2018 didactic\ASM_tools\asm_tools_adriana 2018\npp\multimmm\exemplu>alink mainsuma.obj secondsuma.obj -oPE -subsys console -entry start
ALINK v1.6 (C) Copyright 1998-9 Anthony A.J. Williams.
All Rights Reserved

Loading file mainsuma.obj
Loading file secondsuma.obj
matched Externs
matched ComDefs
Generating PE file mainsuma.exe

D:\_2018 didactic\ASM_tools\asm_tools_adriana 2018\npp\multimmm\exemplu>mainsuma.exe
introduceti a=
2
introduceti b=
4
suma=6
```

Acelasi mecanism se aplica daca trebuie sa prelucram siruri:
Ex: concatenare a 2 siruri definite in data segement
sirurimain.asm

```
bits 32
global start
extern exit, printf
extern concatenare
import printf msvcrt.dll
import exit msvcrt.dll
segment data use32 class=data public
s1 db 'abcdef'
len1 equ $-s1
s2 db '1234'
len2 equ $-s2
s3 times len1+len2+1 db 0
segment code use32 class=code public
start:
; we place all the parameters on the stack
push dword len1
push dword len2
push dword s3
push dword s2
push dword s1

call concatenare

push dword s3
call [printf]
add esp, 4*1

push dword 0
call [exit]
```

Structura stivei:

S1	Esp+4
S2	Esp+8
S3	Esp+12
Len2	Esp+16
Len1	Esp+20

sirurisecond.asm

```
bits 32
segment code use32 class=code public
global concatenare ; export concatenare
concatenare:
```

```
mov esi, [esp+4]
        ;ESI = the offset of the source string (s1)
mov edi, [esp+12]
        ;EDI = the offset of the destination string(s3)
mov ecx, [esp+20]
        ; ECX = len1

cld

repeta:
   lodsb
   stosb
loop repeta

   mov esi, [esp+8] ;s2
   mov ecx, [esp+16] ; len2

   repeta2:
      lodsb
      stosb
   loop repeta2

ret 4*5
```

```
D:\_2018 didactic\ASM_tools\asm_tools_adriana 2018\npp\multimmm\exemplu>sirurimain.exe
abcd1234
D:\_2018 didactic\ASM_tools\asm_tools_adriana 2018\npp\multimmm\exemplu>nasm -f obj sirurimain.asm

D:\_2018 didactic\ASM_tools\asm_tools_adriana 2018\npp\multimmm\exemplu>nasm -f obj sirurisecond.asm

D:\_2018 didactic\ASM_tools\asm_tools_adriana 2018\npp\multimmm\exemplu>alink sirurimain.obj sirurisecond.obj -oPE -subsys console -entry start
ALINK v1.6 (C) Copyright 1998-9 Anthony A.J. Williams.
All Rights Reserved

Loading file sirurimain.obj
Loading file sirurisecond.obj
matched Externs
matched ComDefs
Generating PE file sirurimain.exe

D:\_2018 didactic\ASM_tools\asm_tools_adriana 2018\npp\multimmm\exemplu>sirurimain.exe
abcdef1234
D:\_2018 didactic\ASM_tools\asm_tools_adriana 2018\npp\multimmm\exemplu>
```