

# Tema laborator 13

Pop Mihai-Daniel, Grupa 215/2

## Problema nr. 28 – Asm + Asm

28. Se citeste de la tastatura un sir de caractere (litere mici si litere mari, cifre, caractere speciale, etc). Sa se formeze un sir nou doar cu literele mici si un sir nou doar cu literele mari. Sa se afiseze cele 2 siruri rezultate pe ecran.

Main:

```
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window 1
;28.Se citeste de la tastatura un sir de caractere (litere mici si litere mari, cifre, caractere speciale, etc). Sa se
;formeze un sir nou doar cu literele mici si un sir nou doar cu literele mari. Sa se afiseze cele 2 siruri rezultate pe
;ecran.
1 bits 32
2
3
4 global start
5 extern exit, printf, scanf
6 import exit msvcrt.dll
7 import printf msvcrt.dll
8 import scanf msvcrt.dll
9
10 extern litere_mari
11 extern litere_mici
12
13 segment data use32 class=data
14     format db "Dati sirul de caractere: ",0 ;mesaj
15     format_s db "%s",0 ;formatul pentru citire string caractere
16     format_spatiu db 10,13,0 ;formatul pentru spatiu
17     sir times 100 db 0 ;sirul citit
18     sir_mare times 100 db 0 ;sirul cu litere mari
19     sir_mic times 100 db 0 ;sirul cu litere mici
20
21 segment code use32 class=code
22     start:
23         ;afisarea mesajului pentru citire
24         push format
25         call [printf]
26         add esp, 4
27         ;citirea de la tastatura a unui sir de caractere
28         push sir
```

```
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window 1
28         push sir
29         push format_s
30         call [scanf]
31         add esp, 4*2
32         ;salvarea sirului sursa in esi, sirul destinatie in edi
33         mov esi,sir
34         mov edi,sir_mare
35         ;apelarea funtiei de determinare a literelor mari
36         call litere_mari
37         ;afisarea pe ecran a sirului rezultat
38         push sir_mare
39         call [printf]
40         add esp,4
41         ;trecerea pe o linie noua
42         push format_spatiu
43         call [printf]
44         add esp,4
45         ;salvarea sirului sursa in esi, sirul destinatie in edi
46         mov esi,sir
47         mov edi,sir_mic
48         ;apelarea funtiei de determinare a literelor mici
49         call litere_mici
50         ;afisarea pe ecran a sirului rezultat
51         push sir_mic
52         call [printf]
53         add esp,4
54
55         push dword 0
56         call [exit]
```

## Module:

```
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
1 bits 32
2 segment code use32 public code
3 global litere_mari
4
5 litere_mari:
6     repeta:
7         lodsb      ;incarcarea unui byte in al
8         cmp al,"A" ;verificare daca este litera mare
9         jl next
10        cmp al,"Z"
11        jg next
12        stosb      ;in caz afirmativ punem byte ul in registrul destinatie
13        next:
14        cmp al,""   ;conditie oprire
15        je gata     ;salt oprire
16        jmp repeta  ;salt pentru a trece la urmatorul caracter
17    gata:
18        ret 4 ; in acest caz 4 reprezinta numarul de octeti ce trebuie eliberati de pe stiva (parametrul pasat procedurii)
```

```
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
1 bits 32
2 segment code use32 public code
3 global litere_mici
4
5 litere_mici:
6     repetal:
7         lodsb      ;incarcarea unui byte in al
8         cmp al,"a" ;verificare daca este litera mica
9         jl next1
10        cmp al,"z"
11        jg next1
12        stosb      ;in caz afirmativ punem byte ul in registrul destinatie
13        next1:
14        cmp al,""   ;conditie oprire
15        je gata1    ;salt oprire
16        jmp repetal ;salt pentru a trece la urmatorul caracter
17    gata1:
18        ret 4 ; in acest caz 4 reprezinta numarul de octeti ce trebuie eliberati de pe stiva (parametrul pasat procedurii)
```

## Problema nr. 28 – Asm + C

### Main:

```
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
1 #include <stdio.h>
2 /*28.Se citeste de la tastatura un sir de caractere (litere mici si litere mari, cifre, caractere speciale, etc). Sa se
3 formeze un sir nou doar cu literele mici si un sir nou doar cu literele mari. Sa se afiseze cele 2 siruri rezultate pe
4 ecran.*/
5 char litere_mari(char [],char []);
6
7 char litere_mici(char [],char []);
8
9 int main()
10 {
11     /*Afisarea unui mesaj si citirea sirului de caractere*/
12     char sir[100]="";
13     printf("Introduceti sirul de caractere:\n");
14     scanf("%s",&sir);
15     /*Declararea unui sir pentru rezultat, apelarea funtiei de determinare a literelor mari si afisarea rezultatului*/
16     char rez1[100]="";
17     litere_mari(sir,rez1);
18     printf("%s\n",rez1);
19     /*Declararea unui sir pentru rezultat, apelarea funtiei de determinare a literelor mici si afisarea rezultatului*/
20     char rez2[100]="";
21     litere_mici(sir,rez2);
22     printf("%s\n",rez2);
23 }
```

## Module:

```
1 bits 32
2 segment code use32 public code
3 global _litere_mari
4
5 _litere_mari:
6     push ebp                ;crearea cadrului pe stiva
7     mov ebp,esp
8     mov esi, [ebp+8]        ;preluarea de pe stiva a datelor pentru rezolvarea problemei
9     mov edi, [ebp+12]
10    cld                      ;setarea direction flag-ului
11    repeta:
12        lodsb                ;incarcarea unui byte in al
13        cmp al,"A"           ;verificare daca este litera mare
14        jl next
15        cmp al,"Z"
16        jg next
17        stosb                ;in caz afirmativ punem byte ul in registrul destinatie
18        next:
19        cmp al,""            ;conditie oprire
20        je gata              ;salt oprire
21        jmp repeta           ;salt pentru a trece la urmatorul caracter
22    gata:
23        mov esp,ebp          ;revenirea din procedura
24        pop ebp
25        ret
```

```
1 bits 32
2 segment code use32 public code
3 global _litere_mici
4
5 _litere_mici:
6     push ebp                ;crearea cadrului pe stiva
7     mov ebp,esp
8     mov esi, [ebp+8]        ;preluarea de pe stiva a datelor pentru rezolvarea problemei
9     mov edi, [ebp+12]
10    cld                      ;setarea direction flag-ului
11    repetal:
12        lodsb                ;incarcarea unui byte in al
13        cmp al,"a"           ;verificare daca este litera mica
14        jl next1
15        cmp al,"z"
16        jg next1
17        stosb                ;in caz afirmativ punem byte ul in registrul destinatie
18        next1:
19        cmp al,""            ;conditie oprire
20        je gatal             ;salt oprire
21        jmp repetal          ;salt pentru a trece la urmatorul caracter
22    gatal:
23        mov esp,ebp          ;revenirea din procedura
24        pop ebp
25        ret
```