

Tema Laborator 8

Suciu Radu – Grupa 216/2

Problema 24 – Apeluri funcții de sistem

Se citesc de la tastatura doua numere a si b. Sa se calculeze valoarea expresiei $(a/b)*k$, k fiind o constanta definita in segmentul de date. Afisati valoarea expresiei (in baza 2).

```
bits 32

global start

extern exit,printf,scanf

import exit msvcrt.dll
import printf msvcrt.dll
import scanf msvcrt.dll

segment data use32 class=data

    format db "%d", 0
    amesaj db "a = ", 0
    bmesaj db "b = ", 0
    mesaj db "Valoarea expresiei (a/b)*k este: ", 0
    mesaj_eroare db "b trebuie sa fie diferit de 0", 0
    a dw 0
    a1 dw 0
    b dw 0
    b1 dw 0
    k dw 2
    rezultat times 33 db 0

segment code use32 class=code

start:

    ; afisare: a =
    push dword amesaj
    call [printf]
    add esp, 4 * 1

    ;citire valoare a de la tastatura
    push dword a
    push dword format
    call [scanf]
    add esp, 4 * 2

    ; afisare: b =
```

```

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

;citire valoare b de la tastatura
push dword b
push dword format
call [scanf]
add esp, 4 * 2
cmp dword [b], 0
je eroare

;afisare mesaj "Valoarea expresiei (a/b)*k este: "
push dword mesaj
call [printf]
add esp, 4 * 1

;pun in EDX:EAX valoarea lui a
mov ax, 0
mov ax, [a]
mov dx, 0 ; DX:AX = a

div word [b] ; AX = DX:AX / [b] = a/b

mul word [k] ; DX:AX = AX * [k]
push dx
push ax
pop eax

;transformare in baza 2
mov ecx, 32
mov edi, rezultat
mov ebx, 80000000h ; masca pentru primul bit
repetă:
    test eax, ebx
    jnz bit1

    ; daca bitul e 0
    mov byte [edi], '0'
    jmp bit0 ; daca bitul e zero

bit1:
    ;bitul e 1
    mov byte [edi], '1'

bit0:

```

```

        inc edi

        shr ebx, 1 ; se muta spre dreapta cu o pozitie toti bitii din ebx
    loop repeta

;afisare rezultat
push rezultat
call [printf]
add esp, 4 * 2
jmp final

eroare:
push mesaj_eroare
call [printf]
add esp, 4 * 1

final:
push dword 0
call [exit]

```

Problema 7 - fisiere

Se da un fisier text. Sa se citeasca continutul fisierului, sa se determine litera mica (lowercase) cu cea mai mare frecventa si sa se afiseze acea litera, impreuna cu frecventa acesteia. Numele fisierului text este definit in segmentul de date.

```

bits 32

global start

extern exit, fopen, fclose, fread, printf

import printf msvcrt.dll
import fopen msvcrt.dll
import fread msvcrt.dll
import fclose msvcrt.dll
import exit msvcrt.dll

segment data use32 class=data

    nume_fisier db "a.txt", 0 ;numele fisierului care va fi deschis
    mod_acces db "r", 0 ;modul de deschidere a fisierului

    format db "%d", 0
    descriptor_f dd -1
    len equ 100 ; numarul maxim de caractere din fisier
    text times 100 db 0 ; sirul in care se va citi textul din fisier
    fv times 28 db 0 ; vector de frecventa pentru caractere
    afisare_format db "Litera %c apare de %d ori", 0
    fv_maxima dd 0

```

```

formatc db "%c", 0

segment code use32 class=code

start:

    ;fopen(ume, mod_acces)

    push dword mod_acces

    push dword ume_fisier

    call [fopen]

    add esp, 4 * 2


    mov [descriptor_f], eax

    cmp eax, 0 ; verificam daca fisierul este deschis corect

    je fin

    ;eax = fread(text, 1,len, descriptor_f)

    push dword [descriptor_f]

    push dword len

    push dword 1

    push dword text

    call [fread]

    add esp, 4 * 4

    ;eax = numarul de caractere citite

    mov ebx, 0


    mov ecx, eax

    mov esi, text

    cld

    jecxz no_characters

repetă:

    push ecx

    mov eax, 0

    mov edx, 0

    mov ebx, 0

    ;verificam daca caracterul citit e litera mica


    lodsb

    ; AL = caracterul citit

    cmp al, 97

    jnb nu_e_mica

    cmp al, 122

    ja nu_e_mica

    sub al, 97

```

```

        add byte [fv + eax], 1

        ;calcul fv max

        mov edx, [fv + eax]

        cmp edx, [fv_maxima]


        jb nu_e_mai_mare

        mov [fv_maxima], edx ; frecventa salvata

        mov ebx, eax ; caracterul salvat

        nu_e_mai_mare:

        ;//////////

        nu_e_mica:

        pop ecx

        loop repeta


        ;printf(format, caracter, frecventa)

        push dword [fv_maxima]

        mov dl, 'a'

        add edx, ebx

        push dword edx

        push dword afisare_format

        call [printf]

        add esp, 4 * 3


        no_characters:

        ;fclose(descriptor_f)

        push dword [descriptor_f]

        call [fclose]

        add esp, 4


        fin:

        push    dword 0

        call    [exit]

```

Conținut fișier : AaAaaabbbcdxxxx

Afișare in fereastră cmd: Litera d apare de 5 ori