МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»



**Лабораторна робота №1-3**

з курсу «Системне програмування»

**Варіант 14**

Виконав студент

групи КНз-2

Чалий Михайло

Перевірив

Мазур В.В.

Львів 2014

# Лабораторна робота 1

Завданная: Навчитись працювати з асемблером

Лістінг t1.asm

; Constants

REC\_WIDTH equ 75

REC\_HEIGHT equ 45

REC\_COLOR equ 5

sseg segment stack

db 256 dup(?)

sseg ends

dseg segment

start\_x dw ?

exit\_message db "Press any key to exit...", "$"

dseg ends

cseg segment

assume ss:sseg, cs:cseg, ds:dseg

start:

jmp main

main:

mov ax,seg dseg

mov ds,ax

; VGA Mode

mov ax,0012h

int 10h

; Mouse

mov ax,0

int 33h

; Show cursor

mov ax,1

int 33h

ms:

mov ax,3 ; Read mouse state

int 33h

test bx,001b ; Test if left button not pressed

jz ms

mov ax,2 ; Hide cursor

int 33h

mov start\_x,cx ; X

mov si,cx

add si,REC\_WIDTH

mov di,dx ; Y

add di,REC\_HEIGHT

; Draw rect

mov ah,0ch ; Pixel output mode

mov bh,0 ; Page

mov al,REC\_COLOR

draw\_line:

mov cx,start\_x ; Start from start\_x

draw\_pixel:

int 10h

inc cx ; Next column

cmp cx,si

jl draw\_pixel

inc dx ; Next row

cmp dx,di

jl draw\_line

; Print exit message

mov ah,09

lea dx,exit\_message

int 21h

; Wait user input

mov ah,08h

int 21h

; Clean view port(FreeDOS)

mov ax,0003h ; Text mode

int 10h

; Exit all good

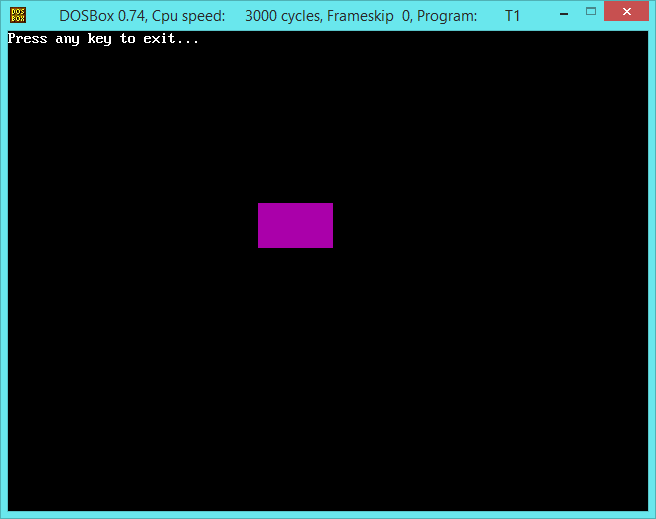
mov ax,4c00h

int 21h

cseg ends

end start

### Результат



# Лабораторна робота 2

title timer

WAIT\_SPAN equ 50 ; Should be microseconds, but in dosbox it looks morelike seconds

TONE equ 1193 ; 1kHZ

sseg segment stack

db 256 dup(?)

sseg ends

dseg segment

ws dd WAIT\_SPAN

dseg ends

cseg segment

assume ss:sseg,cs:cseg,ds:dseg

start: jmp main

main: push ds

mov ax,0

push ax

mov ax,dseg

mov ds,ax

; Enable timer

mov al,10110110b

out 43h,al

mov ax,TONE

out 42h,al

mov al,ah

out 42h,al

; Turn on speaker

in al,61h

or al,11b

out 61h,al

; Wait

mov cx,word ptr ws

mov dx,word ptr ws+2

mov ah,86h

int 15h

; Tuen off speaker

in al,61h

and al,11111100b

out 61h,al

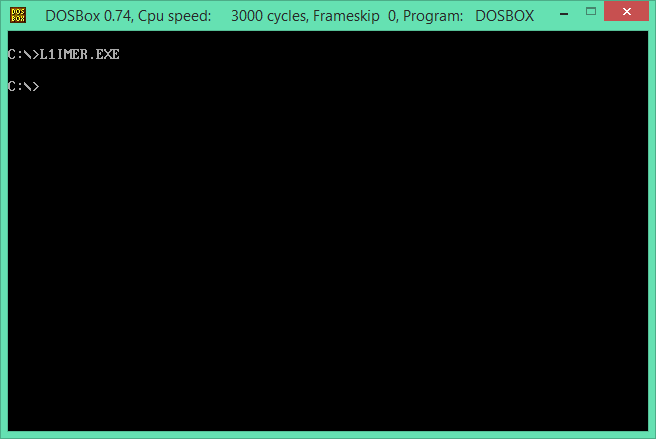
; Exit

mov ax,4c00h

int 21h

cseg ends

end start



# Лабораторна робота 3

Завданная: Работа з файлами. Створити каталог, перейти туди, створити там файл, записати рядок, вийти в рутовий каталог

sseg segment stack

db 256 dup (?)

sseg ends

dseg segment

newcat db '\chaliy',0

newfile db 'mike',0

textbuf db 'Vasiliy'

rootcat db 'c:\',0

error\_msg db 'Unhandled error occured$'

dseg ends

cseg segment

assume cs:cseg,ds:dseg,ss:sseg

start:

jmp main

main:

push ds

mov ax,0

push ax

mov ax,dseg

mov ds,ax

; Create new catalog

mov dx, offset newcat

mov ah,39h

int 21h

jc erop

; Change directory to new catalog

mov dx, offset newcat

mov ah,3bh

int 21h

jc erop

; Create new file

mov dx, offset newfile

mov cx,0

mov ah,5bh

int 21h

jc erop

mov bx,ax ; Store handle

; Write textbuf to file

mov dx, offset textbuf

mov cx,7

mov ah,40h

int 21h

jc erop

; Close file

mov ah,3eh

int 21h

jc erop

; Change directory to root

mov dx, offset rootcat

mov ah,3bh

int 21h

jc erop

exit:

mov ax,4c00h ; Exit all good

int 21h

erop:

mov dx, offset error\_msg

mov ah,9

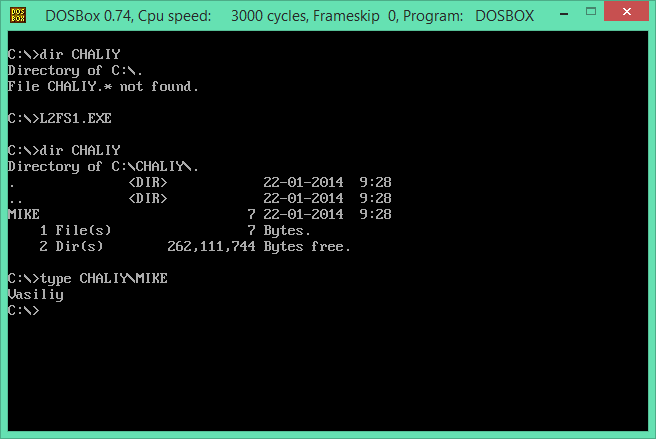
int 21h

jmp exit

cseg ends

end start

### Результат



Завданная: Работа з файлами. Перейти в каталог, прочитати файл, надрукувати, вийти з файла, видлити його, перейти в рут каталог

sseg segment stack ; îïèñ ñòåêó

db 256 dup (?)

sseg ends

dseg segment

cat db '\chaliy',0

file db 'mike',0

buf db 7 dup(?), '$'

rootcat db 'c:\',0

error\_msg db 'Unhandled error occured$'

dseg ends

cseg segment

assume cs:cseg,ds:dseg,ss:sseg

start:

jmp main

main:

push ds

mov ax,0

push ax

mov ax,dseg

mov ds,ax

; Change directory to cat

mov dx,offset cat

mov ah,3bh

int 21h

jc erop

; Open file

mov dx,offset file

mov al,0

mov cx,0

mov ah,3dh

int 21h

jc erop

mov bx,ax

; Read file

mov dx,offset buf

mov cx,8

mov ah,3fh

int 21h

jc erop

; Print

mov ah,9

int 21h

; Close file

mov ah,3eh

int 21h

jc erop

; Delete file

mov dx,offset file

mov ah,41h

int 21h

jc erop

; Change directory to root cat

mov dx,offset rootcat

mov ah,3bh

int 21h

jc erop

; Delete cat

mov dx,offset cat

mov ah,3ah

int 21h

jc erop

exit:

mov ax,4c00h ; Exit all good

int 21h

erop:

mov dx,offset error\_msg

mov ah,9

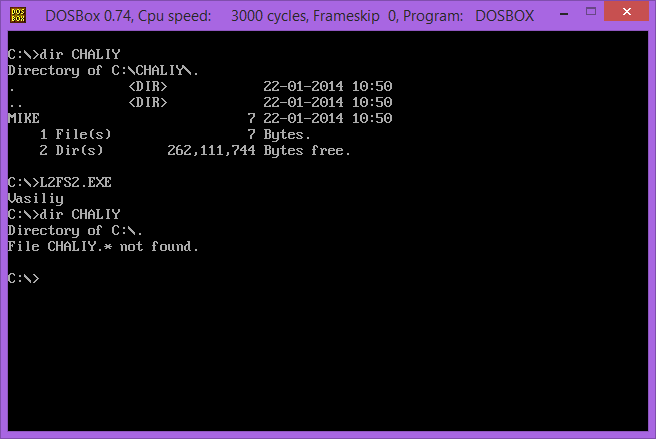
int 21h

jmp exit

cseg ends

end start

### Результат



Висновки

На основі даної розрахункової роботи я дізнався про структуру програми на асемблері, сегменти, дані та команди, закріпив навички з програмування консольних програм, а також роботи з файлами і каталогами.