שימוש בעכבר

# צייר II

קובץ p4, p5

שמרו את התכנית שיצרתם בשיעור שעבר בשם חדש והמשיכו לעבוד בקובץ החדש.

# פסיקת עכבר

**העכבר** – כדי לעבוד עם העכבר במצב גראפי נשתמש בפסיקה **int 33h**.

לפסיקה זאת מספר שימושים:

|  |  |  |
| --- | --- | --- |
| פקודה | פרמטר | אסמבלי |
| אתחול העבודה עם סמן העכבר | ax = 0h | xor ax, ax  int 33h |
| הופעת סמן העכבר על המסך | ax = 1h | mov ax, 1h int 33h |
| הסתרת סמן העכבר | ax = 2h | mov ax, 2h int 33h |
| קריאת מיקום סמן ומצב העכבר | ax = 3h | mov ax, 3h int 33h |

; Boot mouse

mov ax, 0h

int 33h

מקמו פקודות אלו לאחר ציור פלטת הצבעים.

הזיזו את העבר ובדקו שהסמן שלו זז.

; The mouse is visible

mov ax, 1h

int 33h

הריצו את התכנית ובדקו שאם רוצים את העכבר על המסך בנוסף לפלטת הצבעים שלכם.

## העכבר

**מצב הרגיסטרים בהתאם ללחיצת העכבר:**

הפסיקה הקוראת את מיקום העכבר ואת סטטוס הלחיצה עליו:

mov ax,3h

int 33h

הפסיקה מחזירה את הערכים הבאים: -

**bx**מצב הלחיצה על כפתורי העכבר

0 – לא בוצעה לחיצת עכבר

1 – כפתור שמאלי נלחץ

2 – כפתור ימני לחוץ

**dx**מיקום העכבר, עמודה, ציר **y**, ערכים בין 0– 200

**cx**מיקום העכבר, בשורה, ציר **x** ערכים בין 0– 640

(שימו לב, כשאנחנו עובדים במצב גרפי כמות השורות שיש לנו היא 320 בלבד, לכן צריך לבצע התאמה ולחלק את cx בשתיים כדי להגיע למיקום הנכון)

## לולאה ראשית של התכנית – קריאת מצב העכבר

בדיקה האם לחצו על כפתור ימני של העכבר וקריאת מיקום העכבר

; Wait for Mouse

waitMS:

בדיקה של מיקום העכבר על המסך והאם לחצו על הכפתור השמאלי שלו.

בלחיצה על הכפתור השמאלי נצייר פיקסל על המסך.

נחלק את רגיסטר cx ב – 2 כדי למקם אותו נכון במסך

ונפחית מרגיסטר dx 2 כדי שלא יצייר בדיוק על מיקום העכבר ונוכל לראות את הנקודה על המסך.

mov ax,3h

int 33h

cmp bx, 01h ; if left button is down

jne waitMS

; Draw one pixel in Mouse place

shr cx,1 ; Adjust location div in 2 -> line

sub dx, 2 ; sub 2 to see pixel -> column

mov bh,0h

mov al, [color]

xor bh, bh

mov ah, 0Ch

int 10h

מקמו פקודות אלו אחרי הפקודות של אתחול העכבר ולפני הפסיקה הממתינה למקש (ליציאה מהתכנית).

הריצו ובדוק שבהקלדה על הכפתור הימני של המסך תקבלו ציור של פיקסל אחד למסך במיקום של העכבר.

**שמרו את הקובץ בשם חדש המשיכו לעבוד על הקובץ החדש.**

## ציור עם העכבר

נשנה את הפעולה של העכבר:

לחיצה על הכפתור הימני של העכבר תקרא את הצבע עליו הוא נמצא, ולחיצה על כפתור שמאלי של העכבר תצייר על המסך עם הצבע שקראנו.

**bx**מצביע על מצב העכבר:

00b – לא בוצעה לחיצת עכבר

01b – כפתור שמאלי נלחץ

10b – כפתור ימני לחוץ

נשנה את הלולאה הממתינה למצב העכבר:

נפעיל את הפסיקה הקראת את המצב העכבר

נמקם אותו על המסך בהתאם למיקום שנוח לנו

נבדוק את רגיסטר bx

אם הוקש מקש ימני (2) של העכבר נקפוץ לתווית של   
קריאת הצבע - readColor

אם הוקש מקש שמאלי (1) של העכבר נקפוץ לתווית   
של ציור על המסך - draw

אם לא הוקש על העכבר (0) נחזור לתווית של המתנה לעכבר.

waitMS:

mov ax,3h

int 33h

**cmp bx, 10b ; if Right click**

**je readColor**

**cmp bx, 01b ; if Left click**

**je draw**

**jz waitMS**

## שמירת הצבע שקראנו מהמסך למערך הצבעים

נוסיף למערך הצבעים ערך נוסף, 0

ערך זה ישמור את ערך הצבע שנקרא מהמסך ונוכל לקרא אותו כאשר נצייר את הפיקסל למסך.

color dw 2, 3, 4, 14, 15, **0**

את הנקודות על ציר ה-x וציר ה-y, יש לנו מהקריאה של העכבר.

readColor:

draw:

shr cx, 1 ; to position on x

sub dx, 1 ; to position on y

mov ax, [color+5]

xor bh, bh

**mov ah,0Ch**

**int 10h**

jmp waitMS

הריצו את התכנית, כל הזזה של העכבר מציירת על המסך בצבע שחור.

**איבדנו את היכולת לצאת בהקשה על מקש.**

## יציאה מהתכנית בהקשה על Esc

לפני הפסיקה לחזרה למצב טקסט נוסיף תווית – textMod

textMod:

;Retern text mod

mov ax, 2

int 10h

נוסיף ללולאה הבודקת את העבר גם פסיקה הממתינה להקשה על המקלדת,

מתחת לבדיקה של האם ערך העכבר הוא 0 (אפס) jz waitMS -

; Wait for Mouse

waitMS:

mov ax,3h

int 33h

cmp bx, 10 ; if Right click

je readColor

cmp bx, 01 ; if Left click

je draw

**mov ah, 1h ; Wait for key press**

**int 16h**

jz waitMS

**mov ah, 0h ; Read key from keyboard to AL**

**int 16h**

**cmp al , 27 ; if Esc**

**je textMod**

**jmp waitMS**

**הריצו ובדקו האם בהקשה על esc יוצאים מהתכנית.**

## קריאת הצבע במיקום העכבר

readColor:

; read color from screen on mouse place to AL

mov bh, 0h

mov ah, 0dh ; get color of a single pixel.

int 10h

mov [color+5], ax ; AL = pixel color

jmp waitMS

**הריצו ובדקו**

בתכנית שלנו לחיצה על הכפתור השמאלי של העכבר דוגמת את הצבע עליו עומד העכבר.

לחיצה על הכפתור הימני של הכבר מציירת

מקש esc – יוצא מהתכנית.

# הרחבות

**שפרו את התכנית:**

ממתינים ללחיצת העכבר, הציור וקריאת הצבע היא עם לחצן שמאלי של העכבר,

בדיקה האם מיקום העכבר הוא במקום של פלטת הצבעים:

אם כן שמור צבע

אחרת

צייר

הוסיפו צבעים לפלטת הצבעים

# קודה התכנית:

פתיח + משתנים

IDEAL

MODEL small

STACK 100h

**p186**

DATASEG

x\_coordinate dw ? ; place in line

y\_coordinate dw ? ; place in column

color dw 2, 3, 4, 14, 15, 0

x\_begin dw 0 ; Starting point on line

y\_begin dw 25, 50, 75, 100, 125 ; Starting point on column

x\_count db 35 ; loop count line draw

Y\_count db 25 ;loop count column draw

פעולות

CODESEG

proc drawPixel

pusha

xor bh, bh

mov cx, [x\_coordinate] ; place in line

mov dx, [y\_coordinate] ; place in column

mov ax, [color+si]

mov ah, 0ch

int 10h

popa

ret

endp drawPixel

proc DrawRectangle

; Draw square at x\_begin, y\_begin position, size 25\*35

pusha

mov [x\_count], 35 ; Square width

mov [Y\_count], 25 ; Square height

mov ax,[x\_begin] ; Save begin point on X

mov [x\_coordinate],ax

mov ax,[y\_begin + si] ; Save begin point on Y

mov [y\_coordinate],ax

line\_loop:

; line of dots on screen

call drawPixel

inc [x\_coordinate] ; Position the next pixel one step to the left

dec [x\_count]

cmp [x\_count], 0 ; Check if the end of the line loop

jnz line\_loop

mov ax,[x\_begin] ;reset line counters

mov [x\_coordinate], ax

mov [x\_count],35 ;reset column counters

inc [y\_coordinate] ; Position the next pixel one step down

dec [y\_count]

cmp [y\_count], 0 ; Check if the end of the column loop

jnz line\_loop

popa

ret

endp DrawRectangle

תכנית ראשית

start:

mov ax, @data

mov ds, ax

; --------------------------

; Enter Graphic mod 320 \* 200 pixels, 256 colors

mov ax, 13h

int 10h

; Drawing color squares

mov cx, 5

mov si, 0

colorLoop:

call DrawRectangle

add si, 2

loop colorLoop

; ==================== End color palette drawing ============

; Boot mouse

mov ax,0h

int 33h

; The mouse is visible

mov ax,1h

int 33h

; Wait for Mouse

waitMS:

mov ax,3h

int 33h

cmp bx, 10b ; if Right click

je readColor

cmp bx, 01b ; if Left click

je draw

; Wait for key press

mov ah, 1h

int 16h

jz waitMS

; Read key from keyboard to AL

mov ah, 0h

int 16h

cmp al , 27 ; if Esc

je textMod

jmp waitMS

readColor:

; read color from screen on mouse place to AL

mov bh, 0h

mov ah, 0dh ; get color of a single pixel.

int 10h

mov [color+5], ax ; AL = pixel color

jmp waitMS

draw:

; Draw one pixel in Mouse place

shr cx,1 ; Adjust location div in 2 -> line

sub dx, 1 ; sub 2 to see pixel -> column

mov ax,[color+5]

xor bh, bh

mov ah,0Ch

int 10h

jmp waitMS

textMod:

;Retern text mod

mov ax, 2

int 10h

; --------------------------

exit:

mov ax, 4c00h

int 21h

END start