ציור בפיקסלים

# מרובעים ומערכים – צייר I

קובץ p3

בשיעורים הבאים ניצור יחד תכנת צייר. בשיעור זה נשרטט את הרקע של הצייר.

נשרטט מספר מרובעים בצבעים שונים על המסך תוך שימוש בשני מערכים.

צרו תכנית של מעבר למצב גרפי,

המתנה למקש

וחזרה למצב טקסט.

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

mov ax, 13h

int 10h

; Wait for key press

mov ah,0h

int 16h

; return to text mode 80\*20

mov ax, 2

int 10h

## צרו פרוצדורה לציור פיקסל למסך.

והוסיפו לתכנית את הפקודה p186 המאפשרת לעשות משיכה ודחיפה לכל הרגיסטרים יחד.

כדי לצייר פיקסל על המסך יש לספק לאסמבלי את הנתונים הבאים:

מיקום על השורה ציר x,

מיקום בטור, ציר y

וצבע.

**הפרמטרים עבור אינטרפט זה:**

ah = 0ch – קוד הפסיקה

bh = 0 – איפוס רגיסטר bh

cx – עמודה

dx – שורה

ax – צבע

פסיקה המציירת פיקסל למסך:

xor bh, bh

mov cx, [x\_coordinate]

mov dx, [y\_coordinate]

mov ax, [color]

mov ah, 0ch

int 10h

**זמנו את הפרוצדורה מהתכנית הרשאית**

call drawPixel

**הריצו ובדקו.**

IDEAL

MODEL small

STACK 100h

**p186**

DATASEG

x\_coordinate dw 5 ; place in line

y\_coordinate dw 5 ; place in column

color dw 2

CODESEG

proc drawPixel

pusha

xor bh, bh

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

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

mov ax, [color]

mov ah, 0ch

int 10h

popa

ret

endp drawPixel

start:

mov ax, @data

mov ds, ax

call drawPixel

exit:

mov ax, 4c00h

int 21h

END start

# ציור מרובע ⬛

קו הוא אוסף של נקודות ............... על ציר ה – x

וציור של מרובע הוא אוסף של קווים אופקיים.

נצייר את הריבוע כאוסף של קווים בעזרת שתי לולאות:

* לולאה ראשונה לציור קו ולולאה שנייה לציור קווים
* נשנה את הערכים של המיקום על ציר x לערך לא יודע ?
* ואת הערכים של המיקום על ציר y לערך לא יודע?
* נשמור את המיקום של **תחילת ציור** המרובע שלנו על ציר ה – x ועל ציר ה – y.
* ונוסיף שני מונים, מונה ללולאה המונה את הנקודות (המרכיבות קו),
* ומונה ללולאה המונה את הקווים (היוצרים את המרובע).

DATASEG

x\_coordinate dw ? ; place in line

y\_coordinate dw ? ; place in column

color dw 2

x\_begin dw 0 ; Starting point on line

y\_begin dw 30 ; Starting point on column

x\_count db 35 ; loop count line draw

Y\_count db 25 ; loop count column draw

## ניצור פרוצדורה לציור מרובע

CODESEG

Proc DrawRectangle

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

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] ; 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]

dec [y\_count] ; Position the next pixel one step down

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

jnz line\_loop

ret

endp DrawRectangle

קראו לפרוצדורה מהתכנית הראשית

מקמו אותה מתחת לפסיקה המעבירה את המסך למצב גרפי ולפני הפסיקה הממתינה לתו.

call DrawRectangle

הריצו את התכנית ובדקו את התוצאה.

**זכרו 🖰,** כדי לצאת מהתכנית ולחזור למצב טקסט עליכם להקיש על מקש במקלדת.

**שמרו את הקובץ, לשם חדש והמשיכו בקובץ החדש.**

# ציור פלטת הצבעים של הצייר למסך

בתכנת צייר בתפריט מופיעים כמה ריבועים בצבעים שונים, המהווים את פלטת הצבעים מהם בוחר המשתמש את הצבע לציור על המסך.

נוסיף לתכנית שלנו מרובעים בצבעים שונים.

**נשנה** את המשתנה של הצבעים - **[color]** למערך בו נשמור את הערכים של הצבעים שנציג למסך ונוסיף לו ערכים.

color dw 2, 3, 4, 14, 15

ציור המרובעים שלנו מתחיל תמיד באותה עמודה על ציר ה – x (נקודה 0)

אך מיקום השורה (ציר y) משתנה ממרובע למרובע, המרובעים מצוירים למסך זה מתחת לזה.

**נשנה** את המשתנה הקובע את תחילת ציור המרובע על ציר ה – y **[y\_begin]** למערך ונוסיף לו ערכים.

גובה על מרובע שלנו הוא 25 פיקסלים לכן ההפרשים בין הערכים הם 25.

y\_begin dw 25, 50, 75, 100, 125

## רגיסטר si

הוא רגיסטר שניתן להוסיף לשם מערך כדי להצביע על התא במערך אליו נרצה לפנות.

בפעולה של ציור פיקסל נשנה את המשתנה של הצבע **[color]** לשם המערך + רגיסטר si

פעולה זו תאפשר לנו לפנות בכל פעם לפנות לתא הבא במערך.

נשתנה **בפעולה של ציור פיקסל** את המשתנה של הצבע לשם המערך + רגיסטר si

mov ax, [color+si]

נשנה **בפעולה של ציור מרובע** את המשתנה האחראי על מיקום תחילת מרובע הצבעים שלנו על ציר ה – y (השורה בה מתחילים לצייר את המרובע)

mov ax,[y\_begin + si]

## לולאה המציירת את פלטת הצבעים על המסך

בתוכנית הראשית ניצור לולאה המציירת למסך את פלטת הצבעים (המרובעים הצבעונים בצדו השמאלי של המסך, תפריט הצבעים)

בצייר שלנו 5 צבעים ולכן מונה הלולאה הוא 5.

נאפס את רגיסטר si.

; Drawing color squares

mov cx, 5

mov si, 0

colorLoop:

call DrawRectangle

add si, 2

loop colorLoop

הריצו את התכנית ובדקו את התוצאה.

שמרו את הקובץ שלם חדש, המשיכו לעבוד על הקובץ החדש.

* שנו את מיקום הריבועים של פלטת הצבעים
* שנו את גודל הריבועים של פלטת הצבעים
* שנו את הצבעים בפלטה.
* הוסיפו צבעים לפלטת הצבעים.

הרצו את התכנית ובדקו את הערכים. תקנו עד שתגיעו לתוצאה שאתם אוהבים.

# 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 20\*30

pusha

mov [x\_count],35

mov [Y\_count],25

mov ax,[x\_begin]

mov [x\_coordinate],ax

mov ax,[y\_begin + si]

mov [y\_coordinate],ax

line\_loop:

; Print one dot on screen

call drawPixel

inc [x\_coordinate]

dec [x\_count]

cmp [x\_count], 0

jnz line\_loop

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

mov [x\_coordinate], ax

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

inc [y\_coordinate]

dec [y\_count]

cmp [y\_count], 0

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

; Wait for key press

mov ah,0h

int 16h

;Retern text mod

mov ax, 2

int 10h

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

exit:

mov ax, 4c00h

int 21h

END start