

# ΗΥ416 ΓΡΑΦΙΚΑ ΥΠΟΛΟΓΙΣΤΩΝ

Σχεδίαση στον ΗΥ

Π. ΤΣΟΜΠΑΝΟΠΟΥΛΟΥ

ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ & ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

# Συνοπτικά

---

- ▶ Συστήματα αναφοράς συντεταγμένων
- ▶ Αλγόριθμοι σχεδίασης
  - ▶ Γραμμής
  - ▶ Κύκλου
  - ▶ Άλλων καμπύλων

# Συστήματα αναφοράς συντεταγμένων

# Συστήματα αναφοράς συντεταγμένων

---

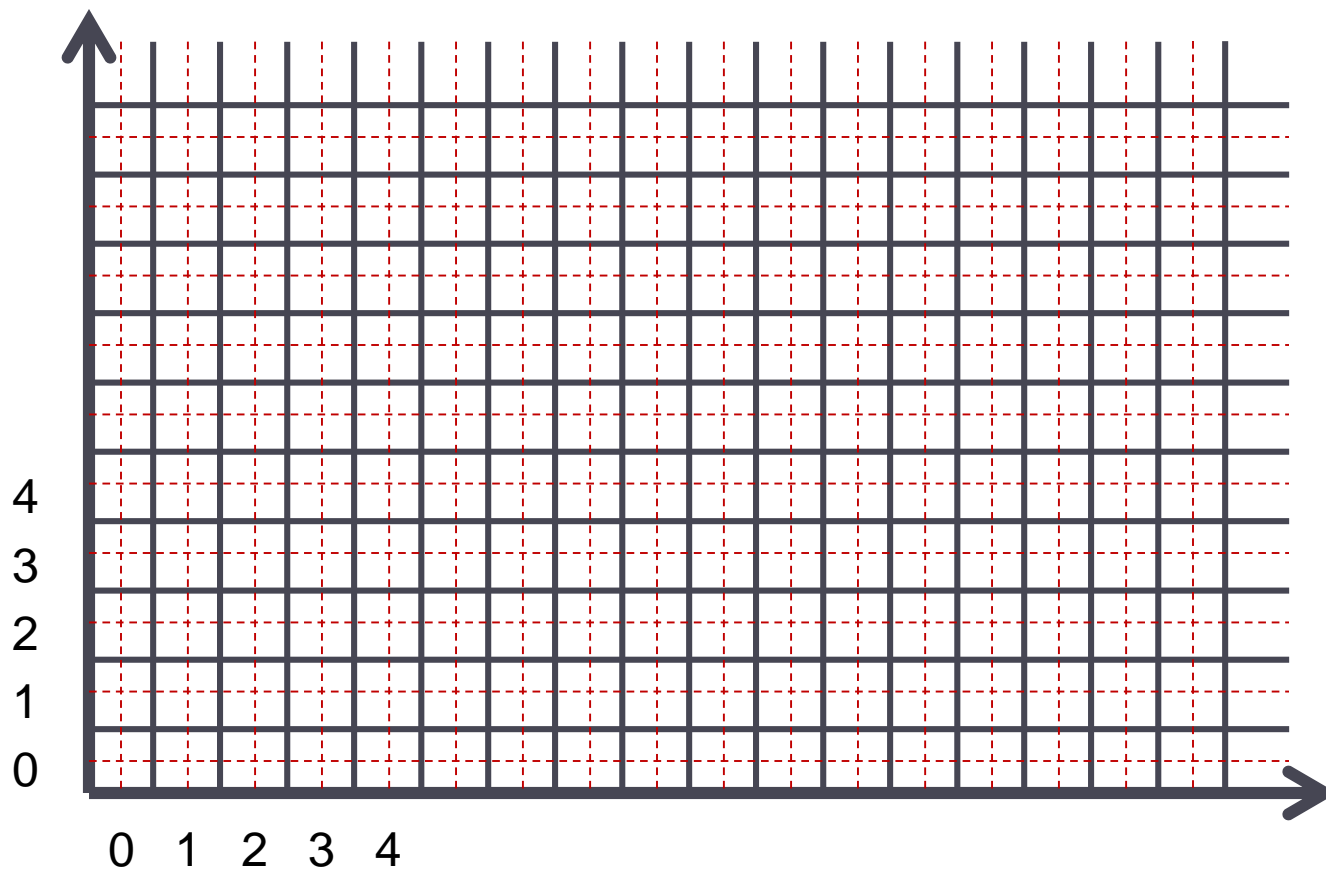
## ▶ Εικόνα →

- ▶ Βολικό καρτεσιανό σύστημα συντεταγμένων (ΣΣ)
- ▶ Γεωμετρική περιγραφή αντικειμένων εικόνας
- ▶ Χρώμα
- ▶ Έκταση συντεταγμένων (ελάχιστες και μέγιστες τιμές σε  $x$ ,  $y$  και  $z$  άξονες) για κάθε αντικείμενο
- ▶ Ορισμός του οριοθετικού πλαισίου (bounding box) = σύνολο των εκτάσεων των συντεταγμένων

## ▶ Συντεταγμένες οθόνης

- ▶ Ακέραιες τιμές
- ▶ Η αρχή των αξόνων βρίσκεται κάτω αριστερά στο παράθυρο (μέρος του πλαισίου) που θα σχηματιστεί η εικόνα.
- ▶ Η μεταφερόμενη εικόνα στην μνήμη της οθόνης, έχει μεταφερθεί στο ΣΣ της οθόνης, δηλ. η αρχή είναι πάνω αριστερά.

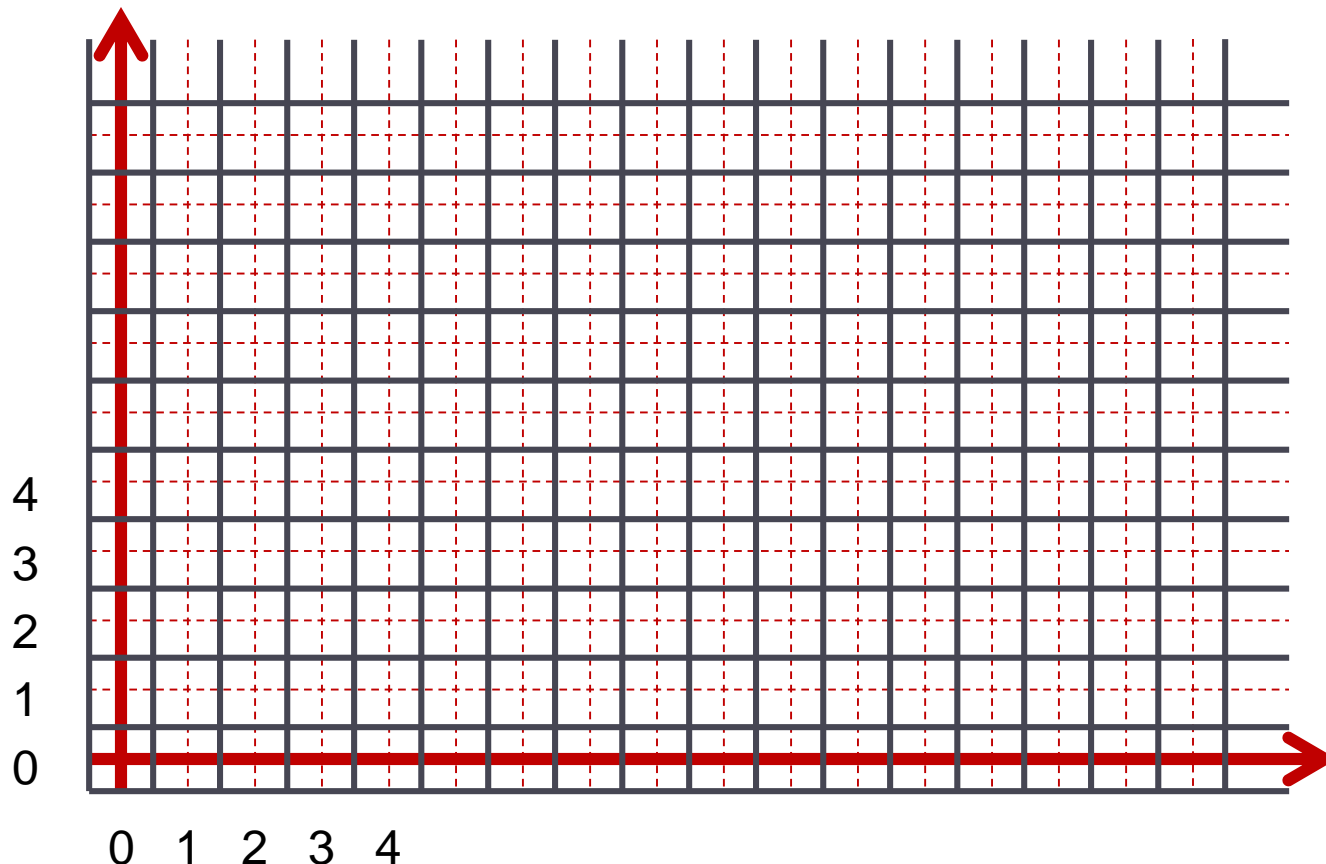
# Συντεταγμένες οθόνης



Οι θέσεις των pixels:

- αναφέρονται σε σχέση με την κάτω-αριστερή γωνία της περιοχής της οθόνης που οριοθετεί το παράθυρο σχεδιασμού.

# Συντεταγμένες οθόνης



Οι θέσεις των pixels:

- αναφέρονται σε σχέση με την κάτω-αριστερή γωνία της περιοχής της οθόνης που οριοθετεί το παράθυρο σχεδιασμού.
- είναι οι συντεταγμένες του κέντρου των αντίστοιχων τετραγώνων

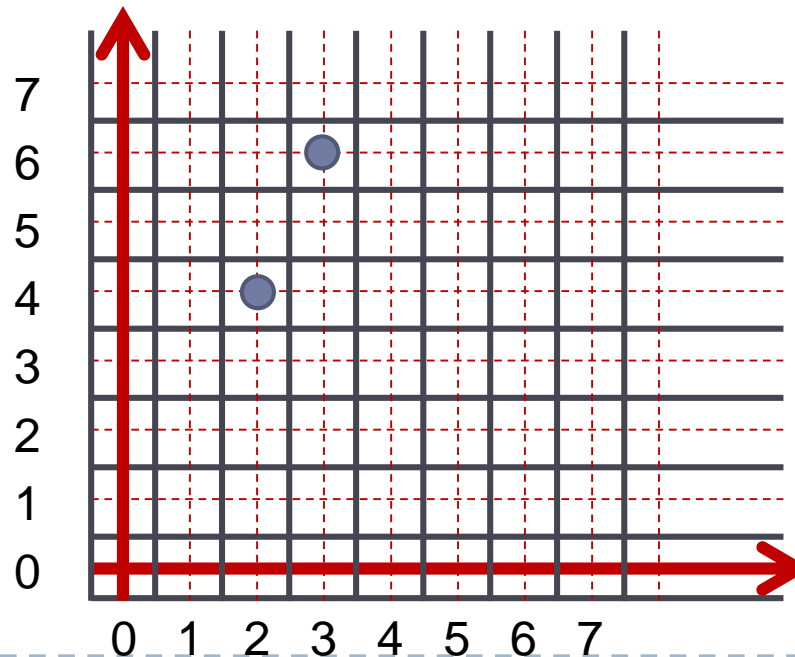
# Συναρτήσεις σημείων της OpenGL

## ▶ glVertex\*\*\*(...)

- ▶ \* = 2, 3, 4 διάσταση του χώρου του σημείου
- ▶ \* = i (integer), s(short), f(float), d(double)
- ▶ \* = κενό ή ν

## ▶ glBegin(GL\_POINTS) , glEnd()

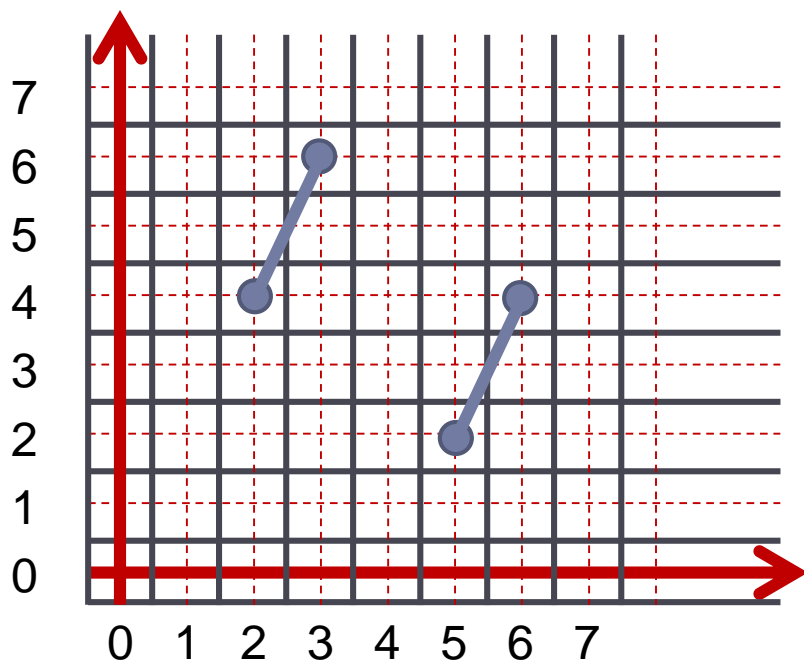
```
glBegin (GL_POINTS);  
glVertex2i (2, 4);  
glVertex2i (3, 6);  
glEnd ();
```



```
int point1[ ] = {2,4};  
int point2[ ] = {3,6};  
glBegin (GL_POINTS);  
glVertex2iv (point1);  
glVertex2iv (point2);  
glEnd ();
```

# Συναρτήσεις γραμμών της OpenGL

- ▶ `glBegin(GL_LINES)` , `glEnd()`

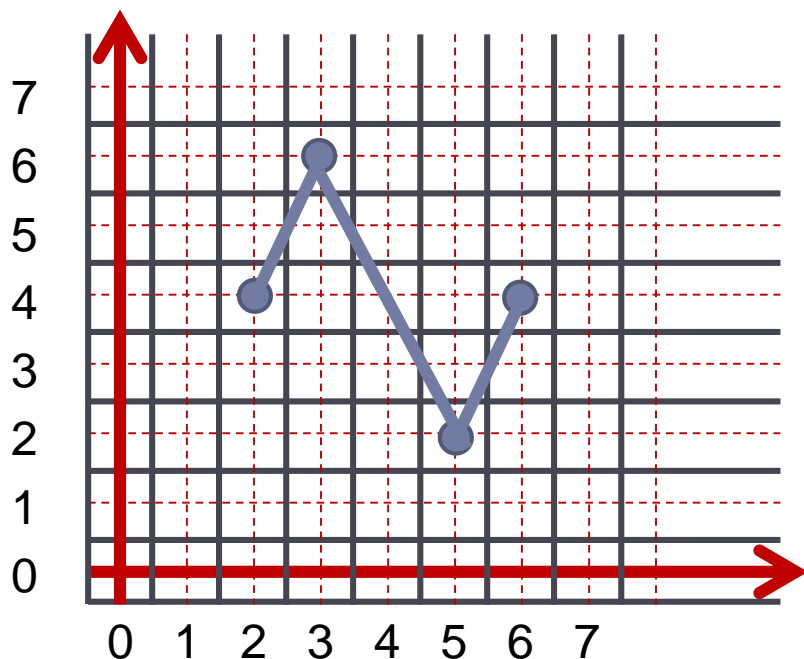


```
int point1[ ] = {2,4};  
int point2[ ] = {3,6};  
int point3[ ] = {5,2};  
int point4[ ] = {6,4};  
glBegin (GL_LINES);  
    glVertex2iv (point1);  
    glVertex2iv (point2);  
    glVertex2iv (point3);  
    glVertex2iv (point4);  
glEnd ();
```



# Συναρτήσεις γραμμών της OpenGL

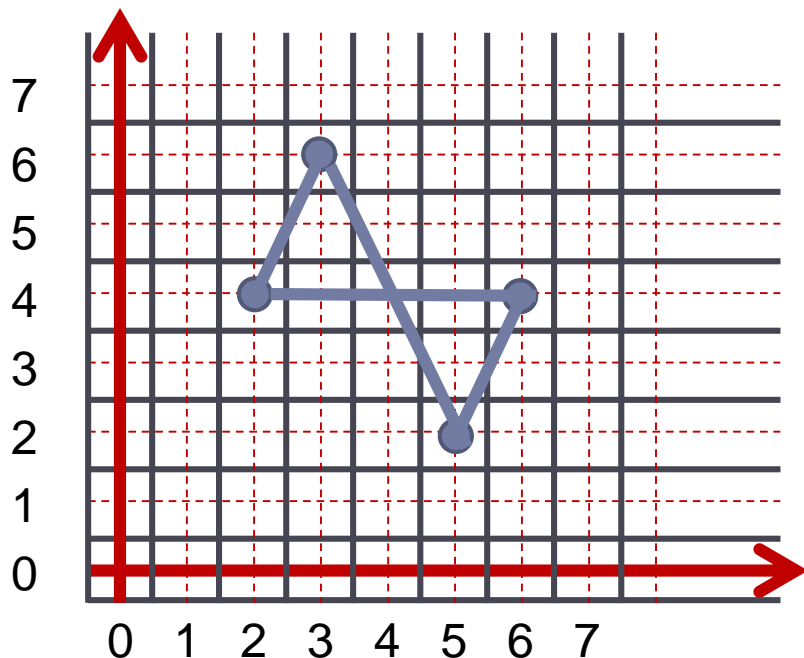
- ▶ `glBegin(GL_LINE_STRIP)` , `glEnd()`



```
int point1[ ] = {2,4};  
int point2[ ] = {3,6};  
int point3[ ] = {5,2};  
int point4[ ] = {6,4};  
glBegin (GL_LINE_STRIP);  
    glVertex2iv (point1);  
    glVertex2iv (point2);  
    glVertex2iv (point3);  
    glVertex2iv (point4);  
glEnd ();
```

# Συναρτήσεις γραμμών της OpenGL

- ▶ `glBegin(GL_LINE_LOOP) , glEnd()`



```
int point1[ ] = {2,4};  
int point2[ ] = {3,6};  
int point3[ ] = {5,2};  
int point4[ ] = {6,4};  
glBegin (GL_LINE_LOOP);  
    glVertex2iv (point1);  
    glVertex2iv (point2);  
    glVertex2iv (point3);  
    glVertex2iv (point4);  
glEnd ();
```

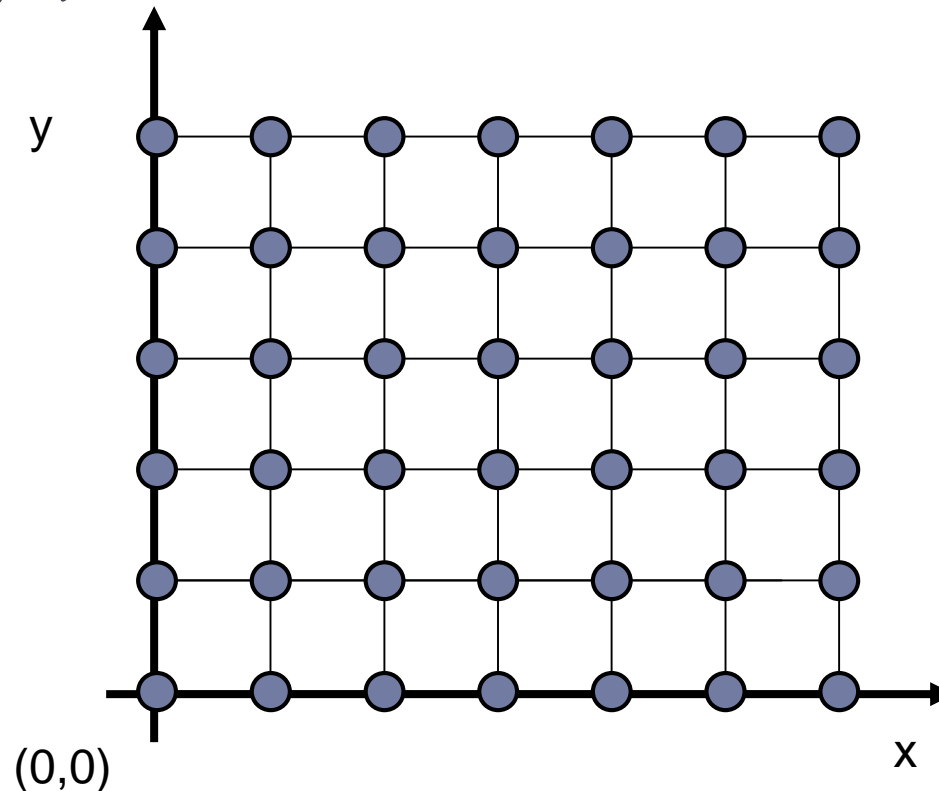
# Αλγόριθμοι σχεδίασης γραμμών

# Pixels

---

## ► Κύρια χαρακτηριστικά

- Θέση  $(i,j)$
- Χρώμα  $(r,g,b)$



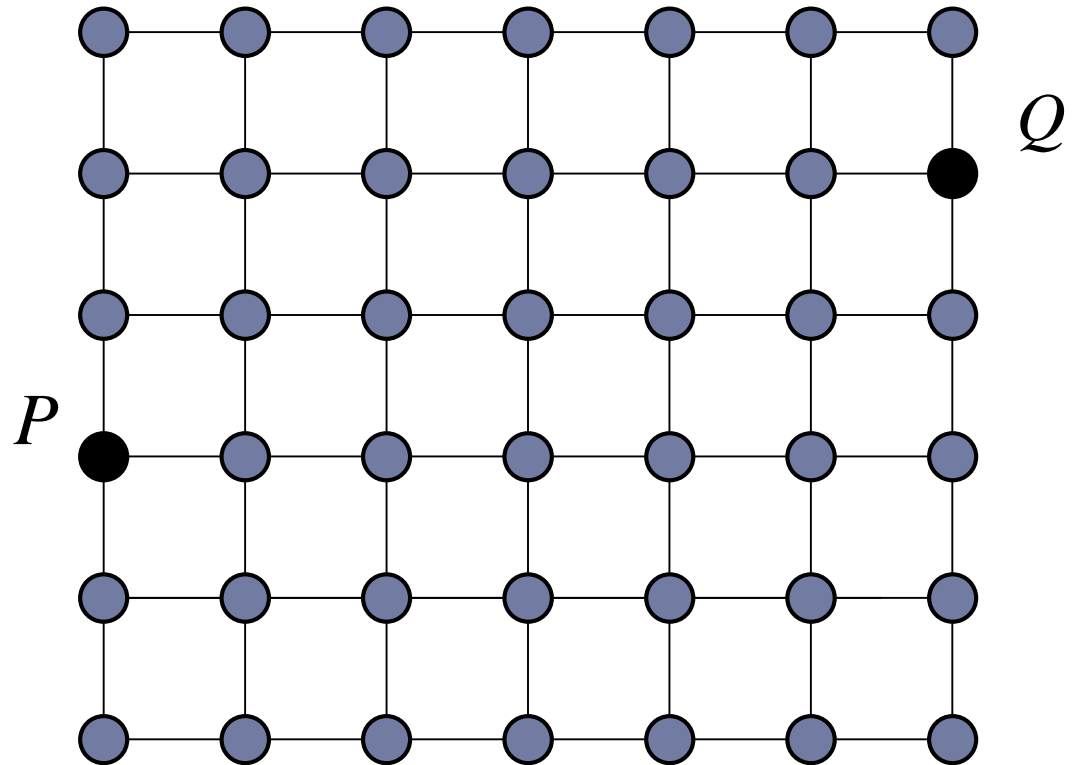
# Αλγόριθμοι

---

- ▶ Digital Differential Analyzer (DDA)
- ▶ Midpoint algorithm

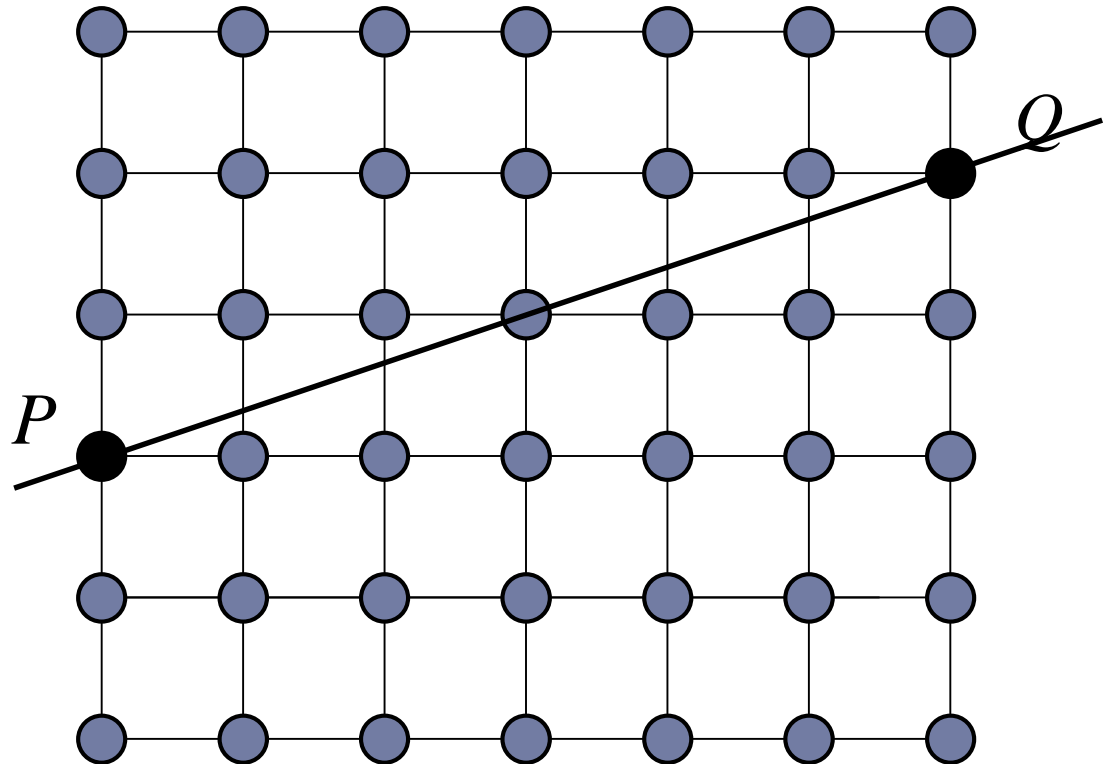
# Πρόβλημα

Δεδομένων σημείων  $(P, Q)$  στην οθόνη (ακέραιες συντεταγμένες), καθορίστε ποια pixels πρέπει να ζωγραφιστούν για να σχηματίσουν μια γραμμή μοναδιαίου πλάτους.



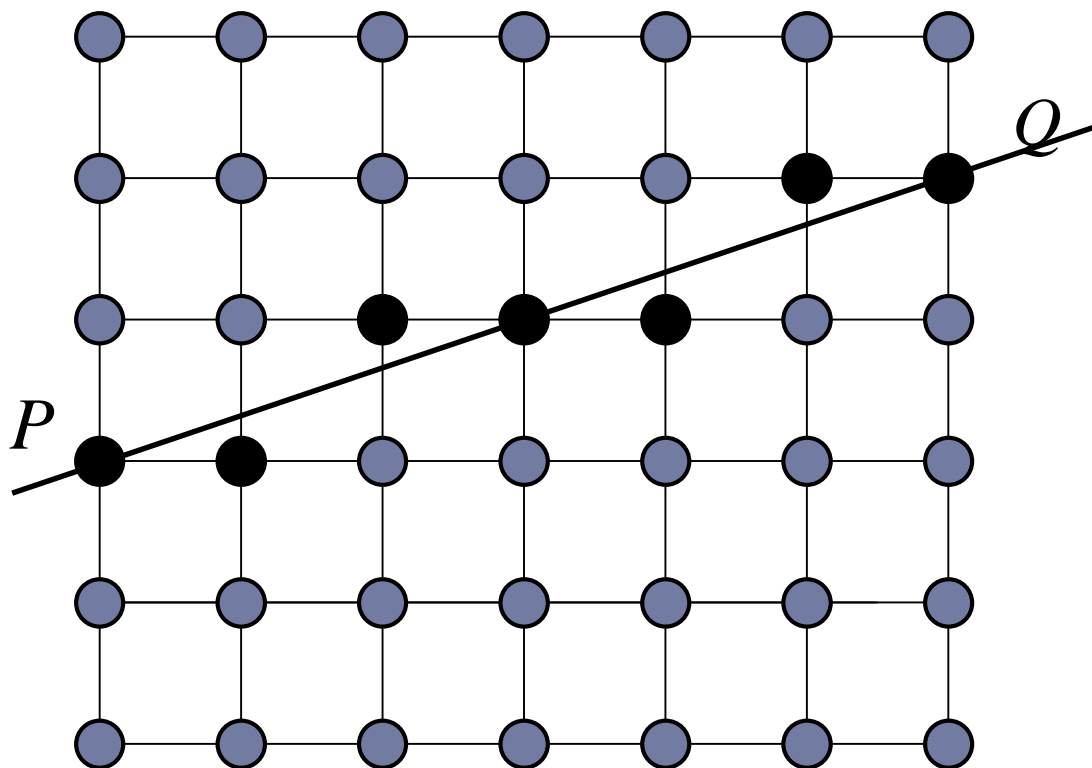
# Πρόβλημα

Δεδομένων σημείων  $(P, Q)$  στην οθόνη (ακέραιες συντεταγμένες), καθορίστε ποια pixels πρέπει να ζωγραφιστούν για να σχηματίσουν μια γραμμή μοναδιαίου πλάτους.



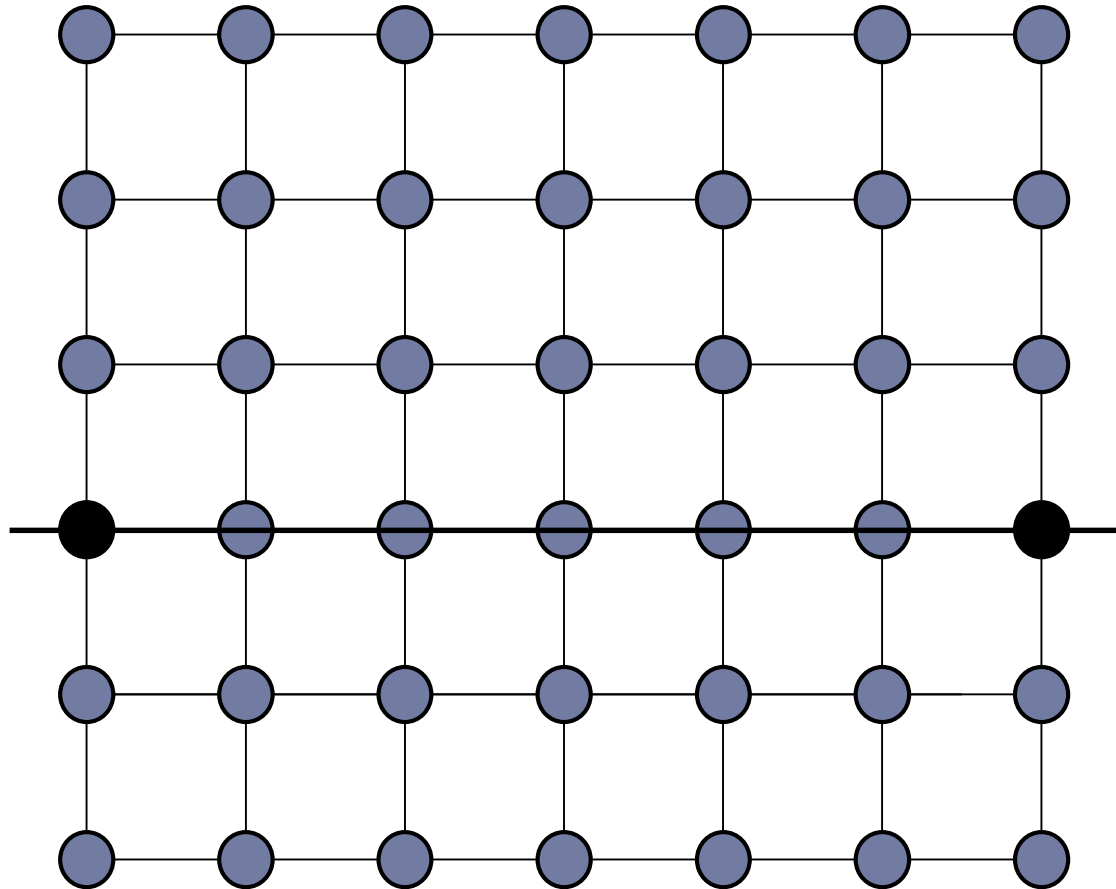
# Πρόβλημα

Δεδομένων σημείων  $(P, Q)$  στην οθόνη (ακέραιες συντεταγμένες), καθορίστε ποια pixels πρέπει να ζωγραφιστούν για να σχηματίσουν μια γραμμή μοναδιαίου πλάτους.





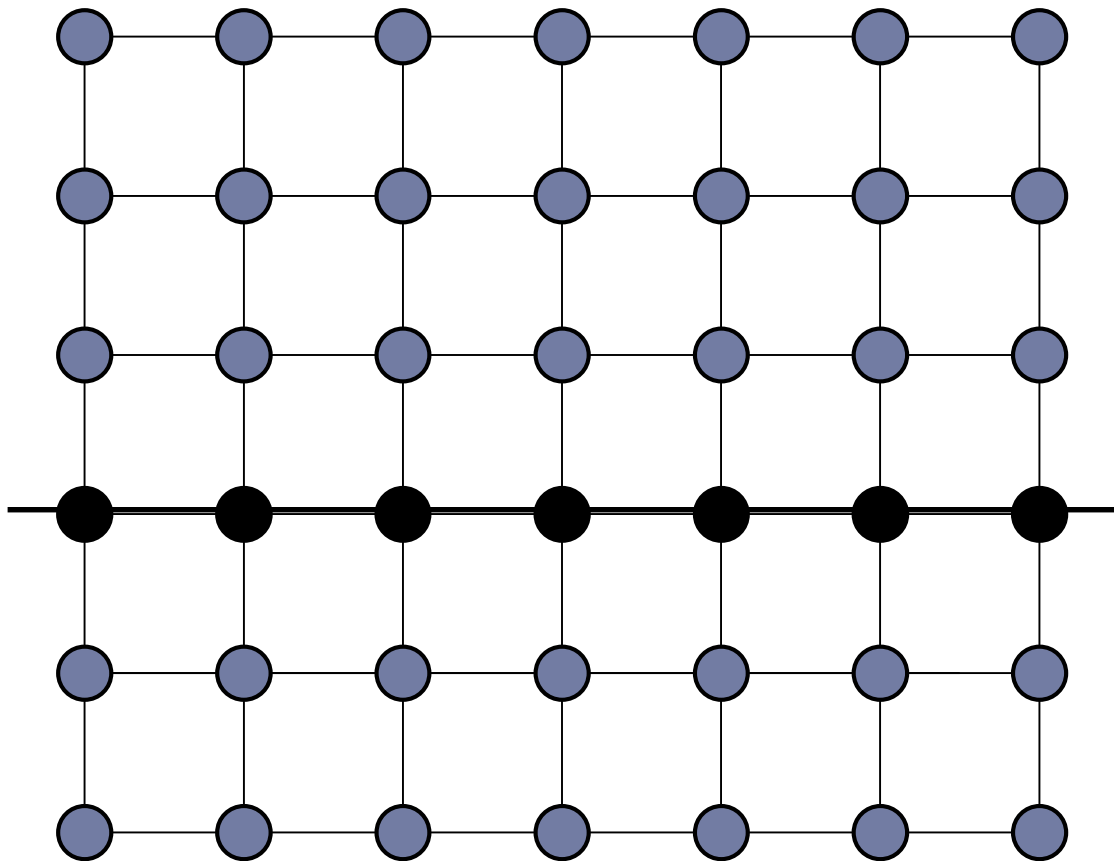
# Ειδικές περιπτώσεις - Οριζόντιες γραμμές



# Ειδικές περιπτώσεις - Οριζόντιες γραμμές

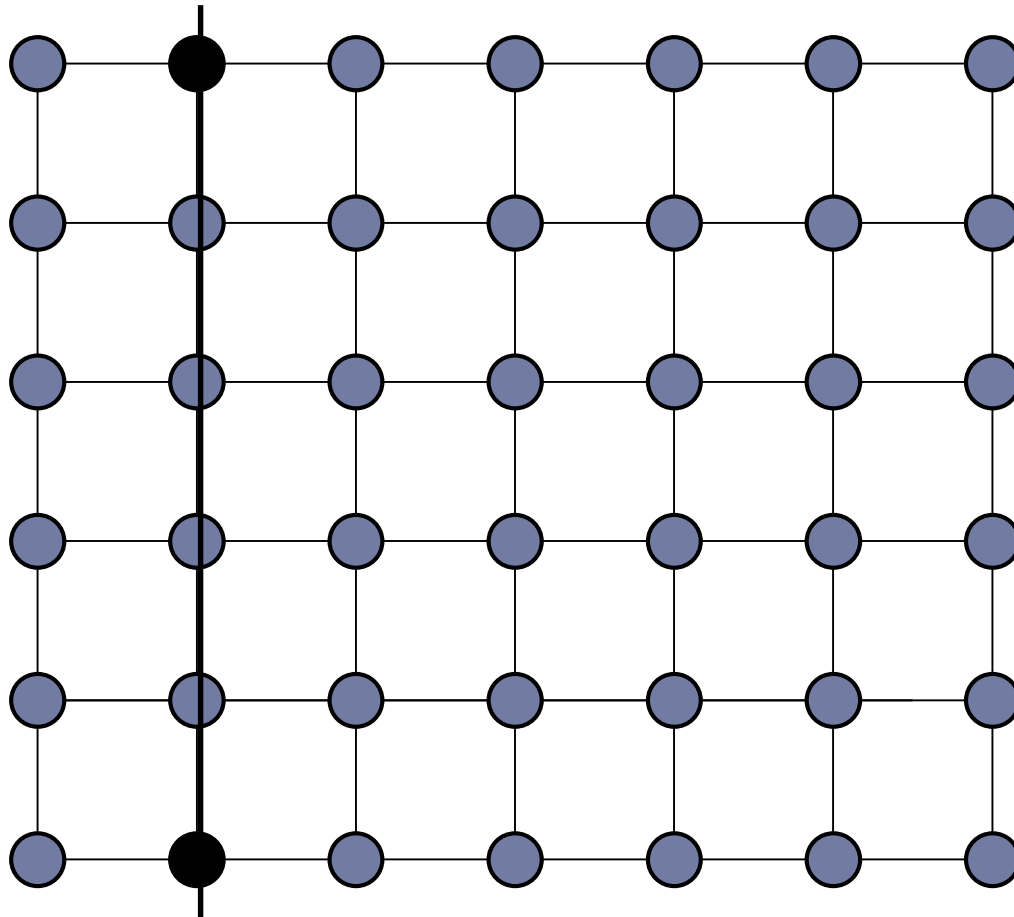
Αύξησε το  $x$  κατά 1

Κράτησε το  $y$  σταθερό



## Ειδικές περιπτώσεις - Κάθετες γραμμές

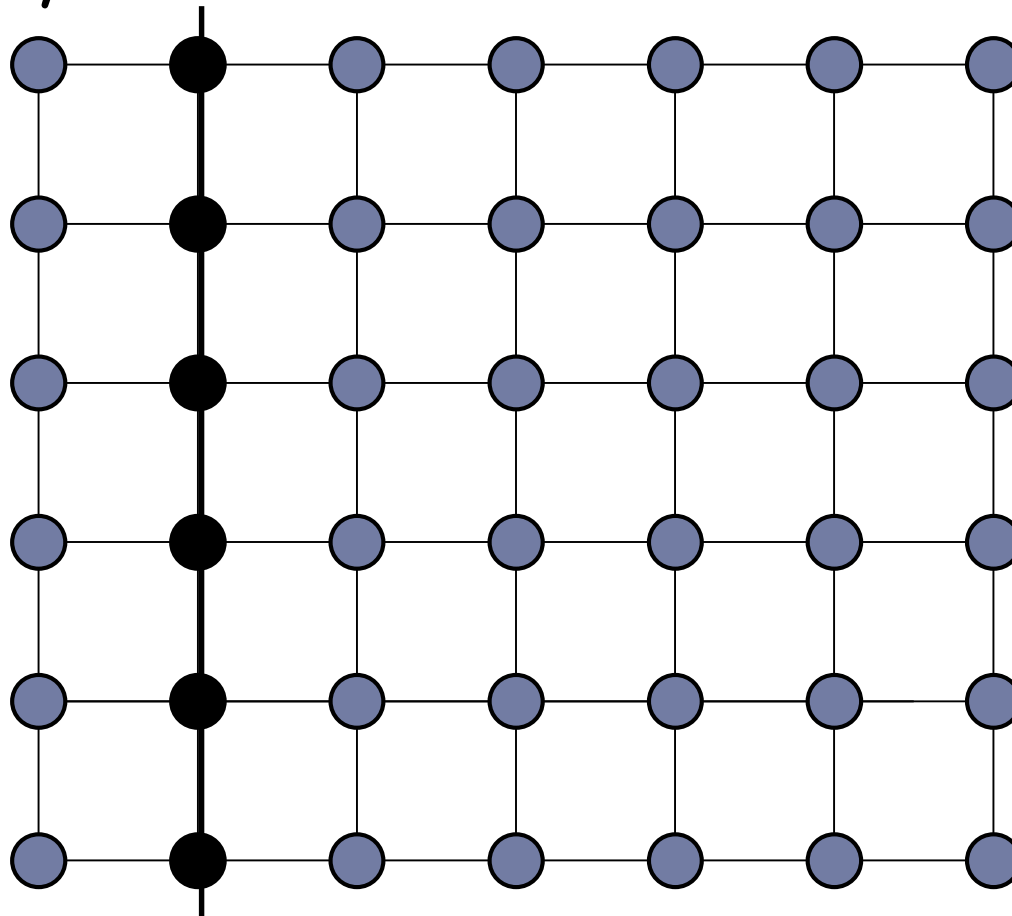
---



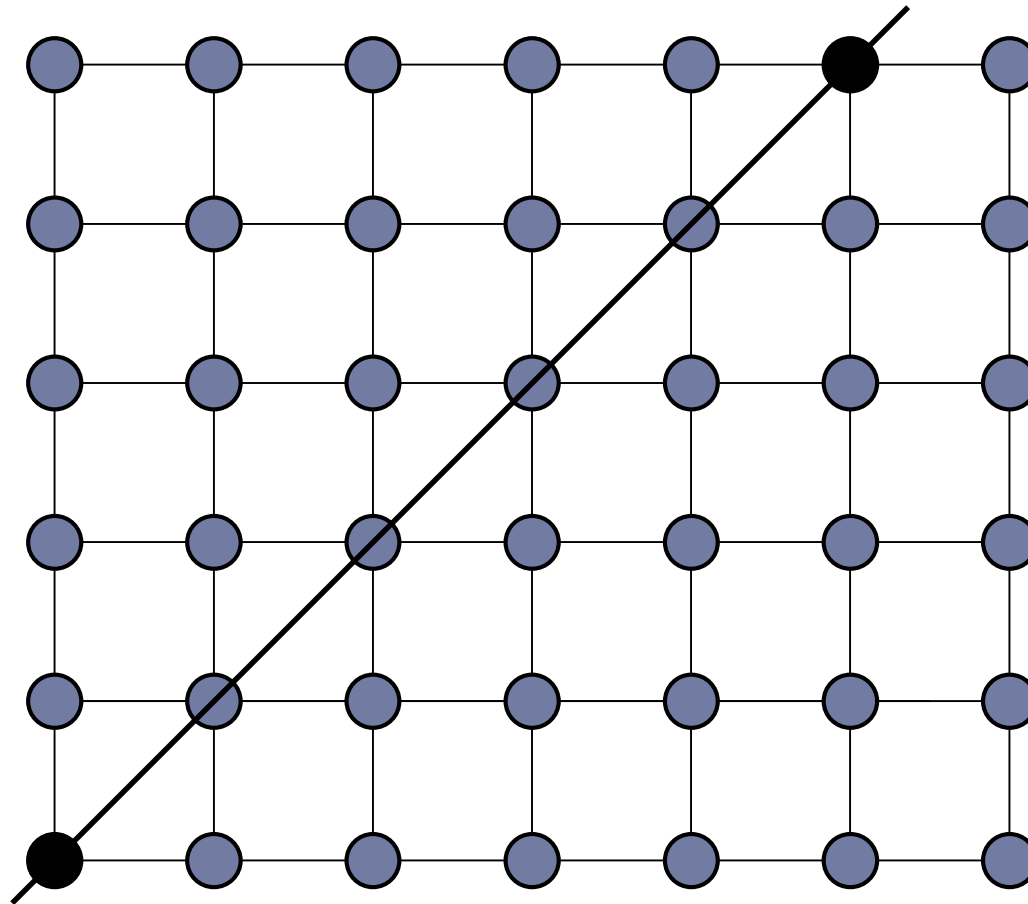
# Ειδικές περιπτώσεις - Κάθετες γραμμές

Κράτησε το  $x$  σταθερό

Αύξησε το  $y$  κατά 1



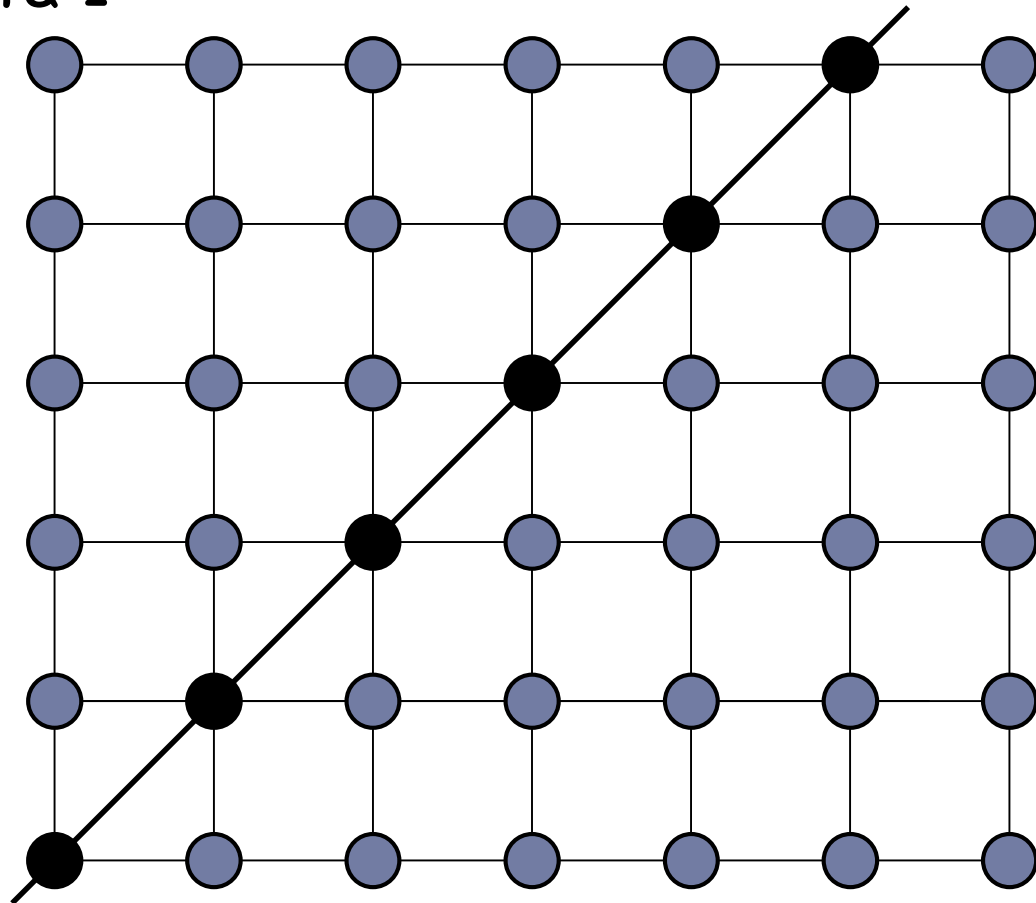
# Ειδικές περιπτώσεις - Διαγώνιες γραμμές



# Ειδικές περιπτώσεις - Διαγώνιες γραμμές

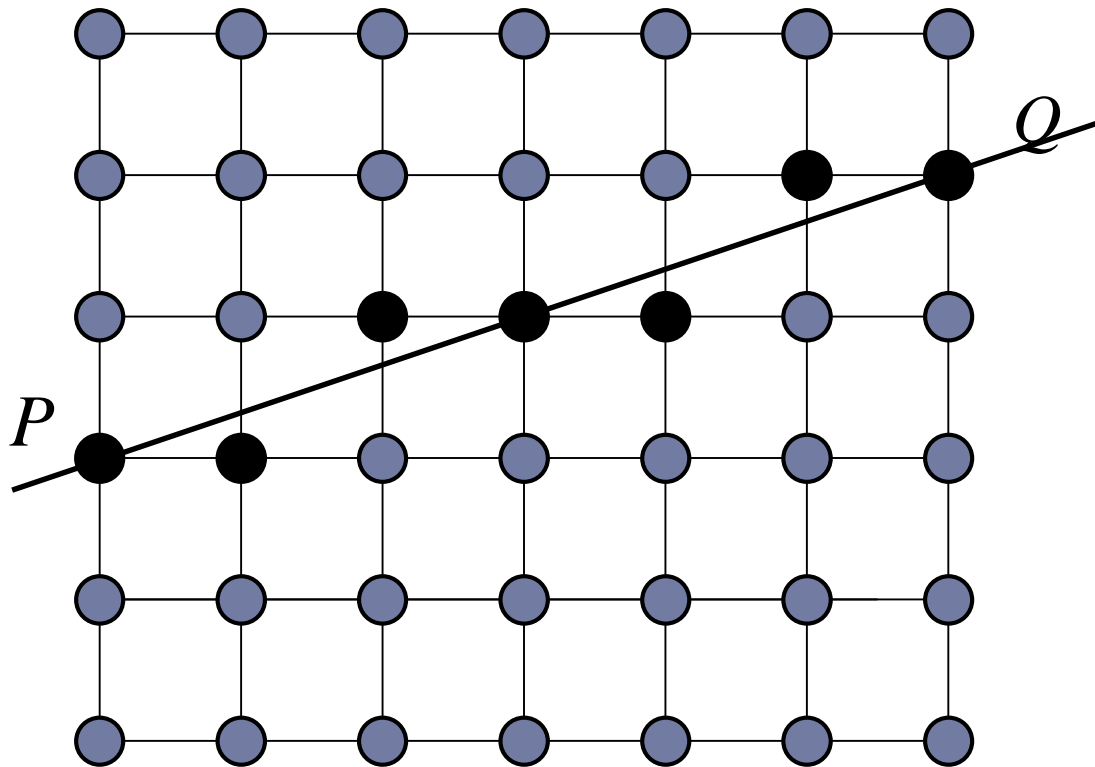
Αύξησε το  $x$  κατά 1

Αύξησε το  $y$  κατά 1



# Τυχαίες γραμμές

---

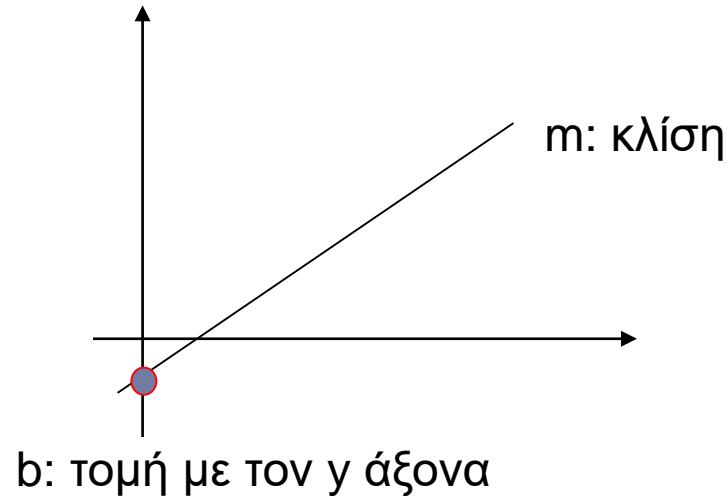


# Τυχαίες γραμμές

---

Εξίσωση γραμμής:

$$y = mx + b$$

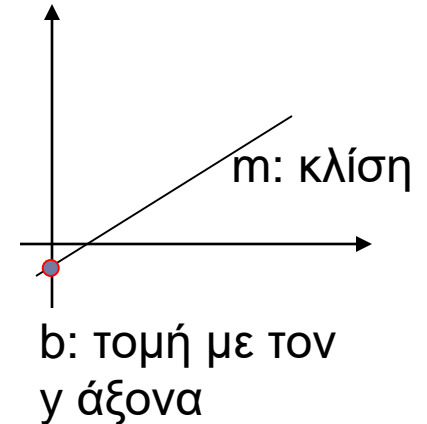




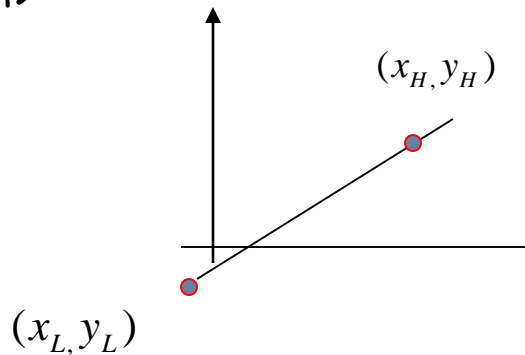
# Τυχαίες γραμμές

Εξίσωση γραμμής:

$$y = mx + b$$



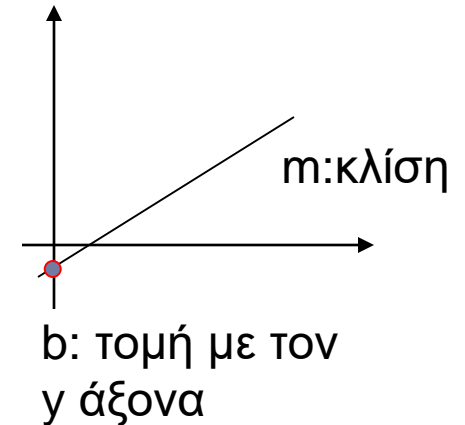
Πως υπολογίζουμε μια γραμμή που περνά από τα σημεία  $(x_L, y_L)$ ,  $(x_H, y_H)$ ?



# Τυχαίες γραμμές

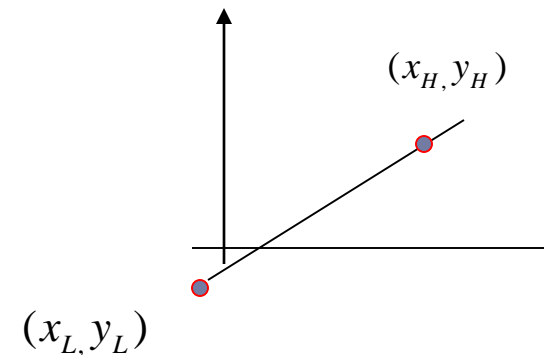
Εξίσωση γραμμής:

$$y = mx + b$$



Πως υπολογίζουμε μια γραμμή που περνά από τα σημεία  
 $(x_L, y_L), (x_H, y_H)$ ?

$$m = \frac{y_H - y_L}{x_H - x_L},$$
$$b = y_L - mx_L$$



# Τυχαίες γραμμές

---

$$y = mx + b$$

- ▶ Υποθέτουμε  $0 < m \leq 1$
- ▶ Με συμμετρία αντιμετωπίζουμε όλες τις άλλες περιπτώσεις
- ▶ Προχωράμε κατά  $x$  και καθορίζουμε τι συμβαίνει κατά  $y$  (ποιο pixel θα ζωγραφιστεί)

$$\Delta y = m \Delta x$$

# Κριτήρια καλού αλγόριθμου σχεδίασης ευθύγραμμου τμήματος

---

- ▶ Σταθερό πάχος, ανεξάρτητο κλίσης
- ▶ Όχι κενά (συνεκτικότητα)
- ▶ Σχεδιασμένα pixels όσο το δυνατόν πλησιέστερα στην μαθηματική πορεία της γραμμής
- ▶ Ταχύτητα

# Digital Differential Analyzer

(ψηφιακός διαφορικός αναλυτής)

# Digital Differential Analyzer

---

Start from  $(x_L, y_L)$  and draw to  $(x_H, y_H)$  where  $x_L < x_H$

$(x_0, y_0) = (x_L, y_L)$

For  $(i = 0; i \leq x_H - x_L; i++)$

    DrawPixel  $(x_i, \text{Round}(y_i))$

$x_{i+1} = x_i + 1$

$y_{i+1} = m x_{i+1} + b$

# Digital Differential Analyzer

---

Απλός αλγόριθμος:

- επέλεξε μοναδιαία διαστήματα στην μια διάσταση
- βρες το πλησιέστερο στην γραμμή `pixel` στην άλλη διάσταση.

# Digital Differential Analyzer (DDA)

Start from  $(x_L, y_L)$  and draw to  $(x_H, y_H)$  where  $x_L < x_H$

$$(x_0, y_0) = (x_L, y_L)$$

For  $(i = 0; i \leq x_H - x_L; i++)$  // sample unit intervals in x coordinate

DrawPixel  $(x_i, \text{Round}(y_i))$  // find the nearest pixel for each sampled point.

$$x_{i+1} = x_i + 1$$

$$y_{i+1} = m(x_i + 1) + b$$



# DDA

---

Start from  $(x_L, y_L)$  and draw to  $(x_H, y_H)$  where  $x_L < x_H$

$(x_0, y_0) = (x_L, y_L)$

For  $(i = 0; i \leq x_H - x_L; i++)$

DrawPixel  $(x_i, \text{Round}(y_i))$

$x_{i+1} = x_i + 1$

$y_{i+1} = m x_i + m + b$

# DDA

---

Start from  $(x_L, y_L)$  and draw to  $(x_H, y_H)$  where  $x_L < x_H$

$(x_0, y_0) = (x_L, y_L)$

For  $(i = 0; i \leq x_H - x_L; i++)$

DrawPixel  $(x_i, \text{Round}(y_i))$

$x_{i+1} = x_i + 1$

$y_{i+1} = m x_i + b + m$

# DDA

---

Start from  $(x_L, y_L)$  and draw to  $(x_H, y_H)$  where  $x_L < x_H$

$(x_0, y_0) = (x_L, y_L)$

For  $(i = 0; i \leq x_H - x_L; i++)$

DrawPixel  $(x_i, \text{Round}(y_i))$

$x_{i+1} = x_i + 1$

$y_{i+1} = m x_i + b + m$

# DDA

---

Start from  $(x_L, y_L)$  and draw to  $(x_H, y_H)$  where  $x_L < x_H$

$(x_0, y_0) = (x_L, y_L)$

For  $(i = 0; i \leq x_H - x_L; i++)$

DrawPixel  $(x_i, \text{Round}(y_i))$

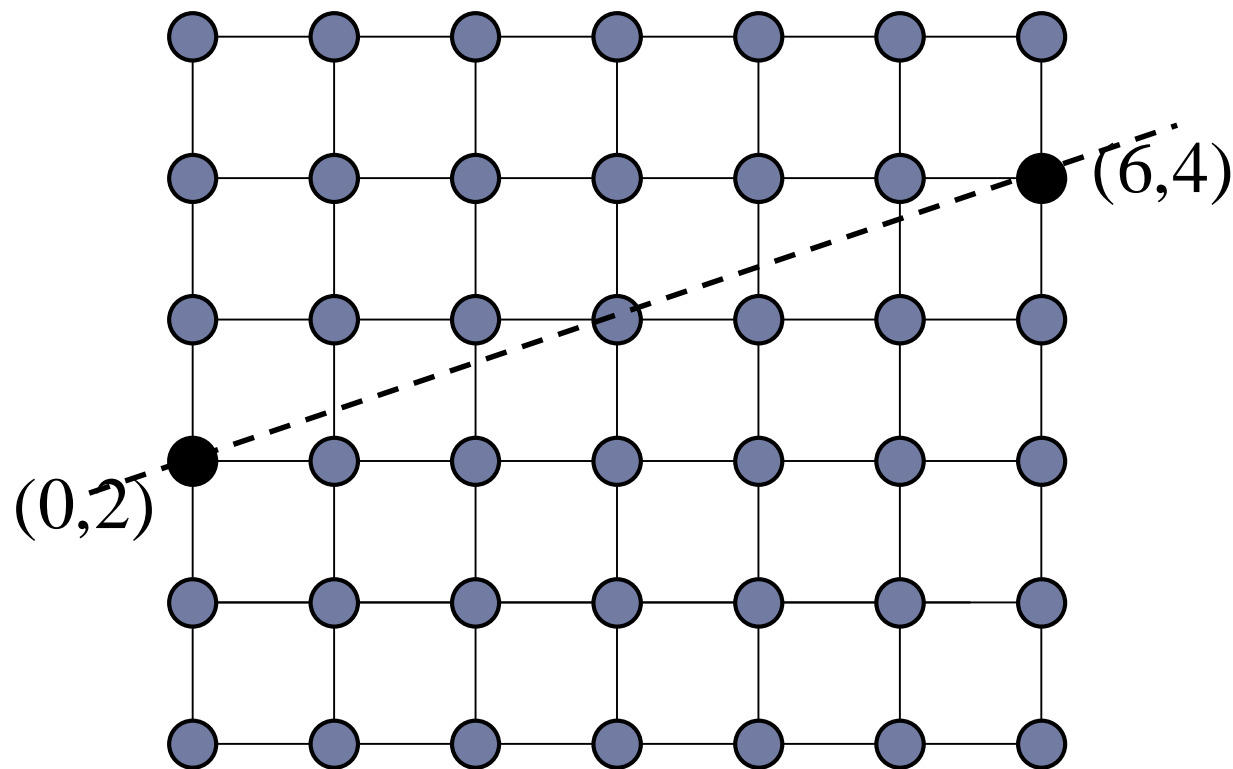
$x_{i+1} = x_i + 1$

$y_{i+1} = y_i + m$

# DDA - Παράδειγμα

---

$$y = mx + b$$

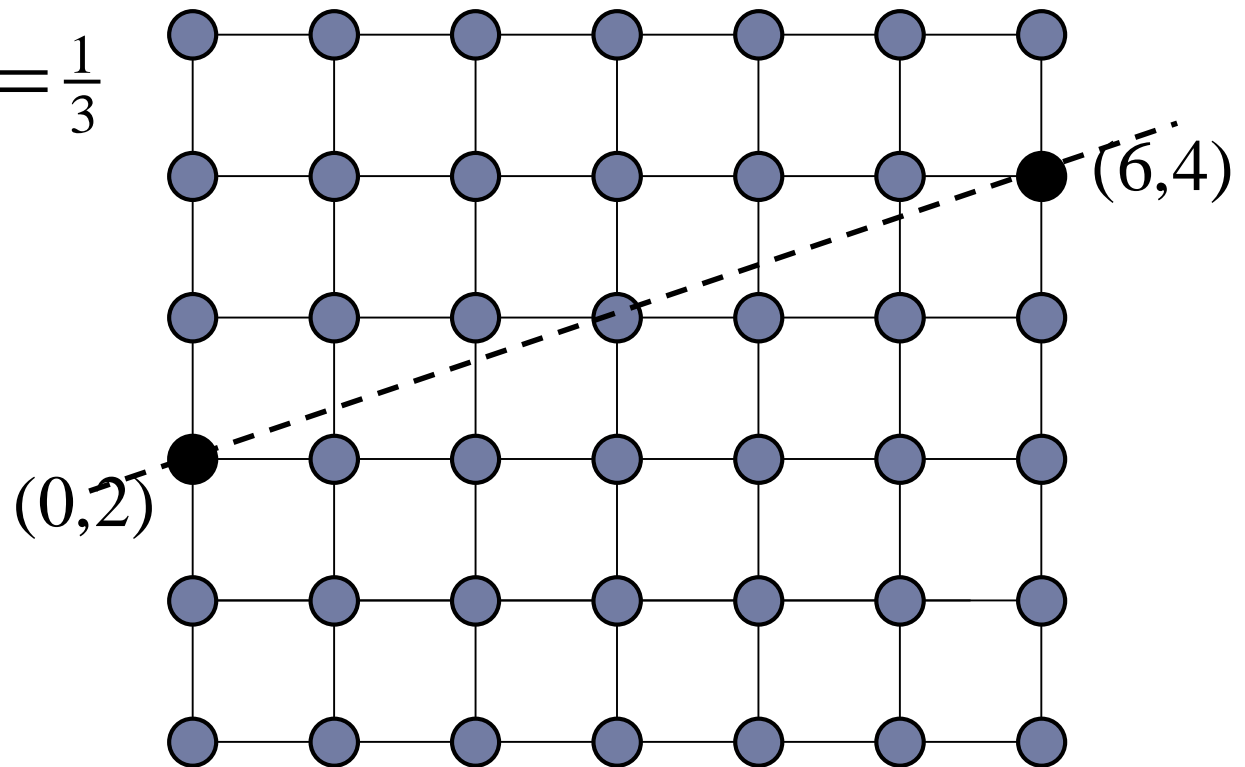


# DDA - Example

---

$$y = mx + b$$

$$m = \frac{y_H - y_L}{x_H - x_L} = \frac{4 - 2}{6 - 0} = \frac{1}{3}$$



# DDA - Example

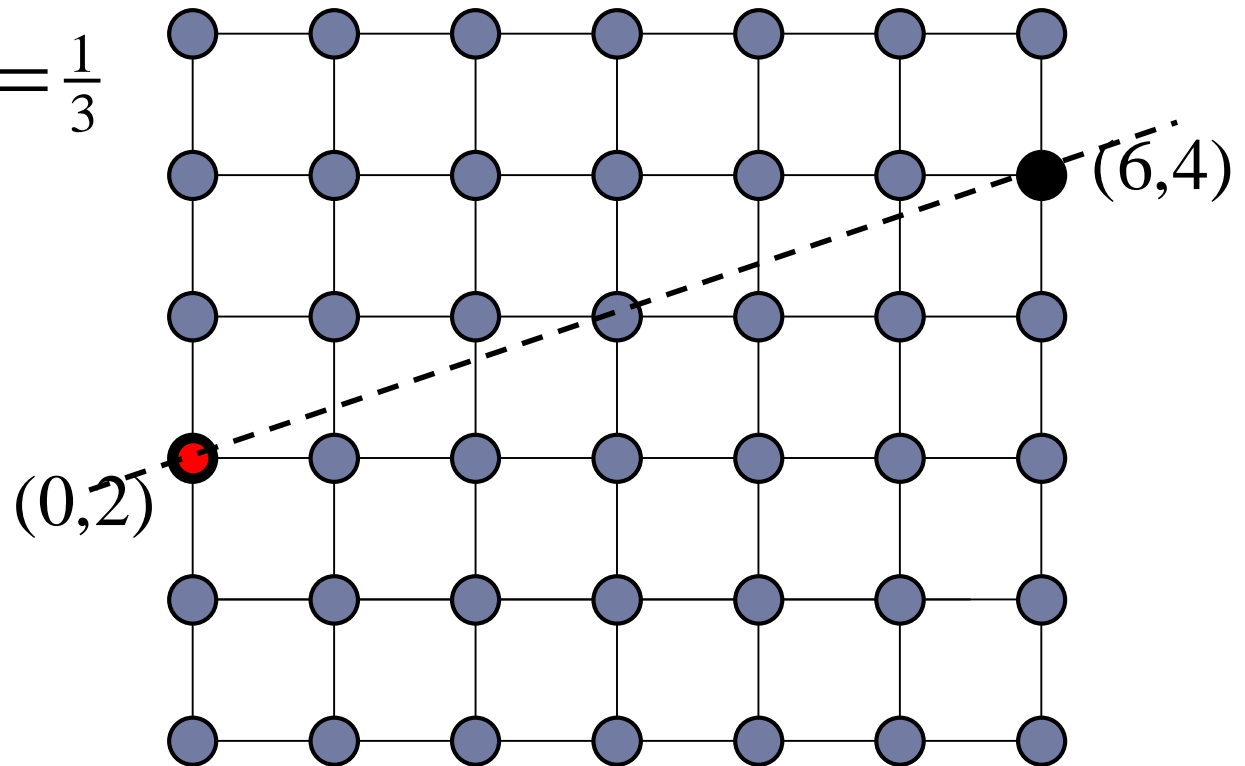
---

$$y = mx + b$$

$$m = \frac{y_H - y_L}{x_H - x_L} = \frac{4 - 2}{6 - 0} = \frac{1}{3}$$

$$x = 0$$

$$y = 2$$



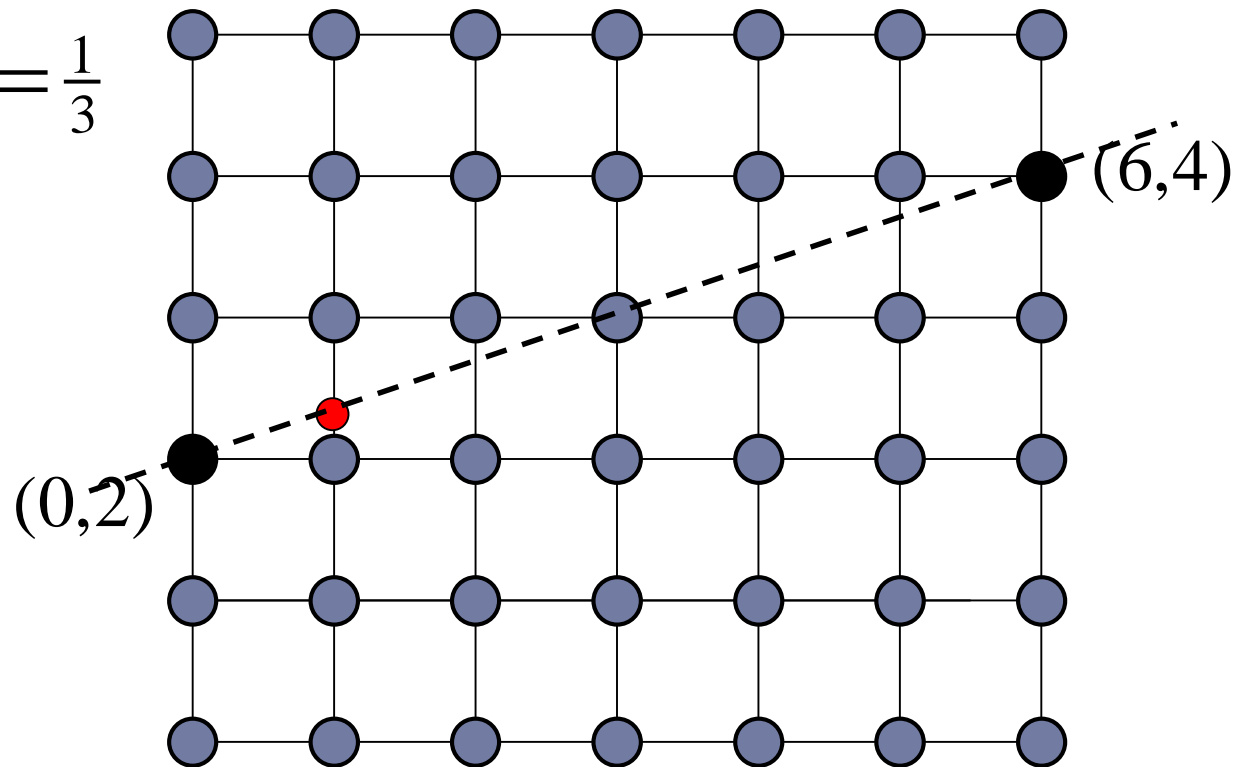
# DDA - Example

$$y = mx + b$$

$$m = \frac{y_H - y_L}{x_H - x_L} = \frac{4 - 2}{6 - 0} = \frac{1}{3}$$

$$x = 1$$

$$y = \frac{7}{3}$$





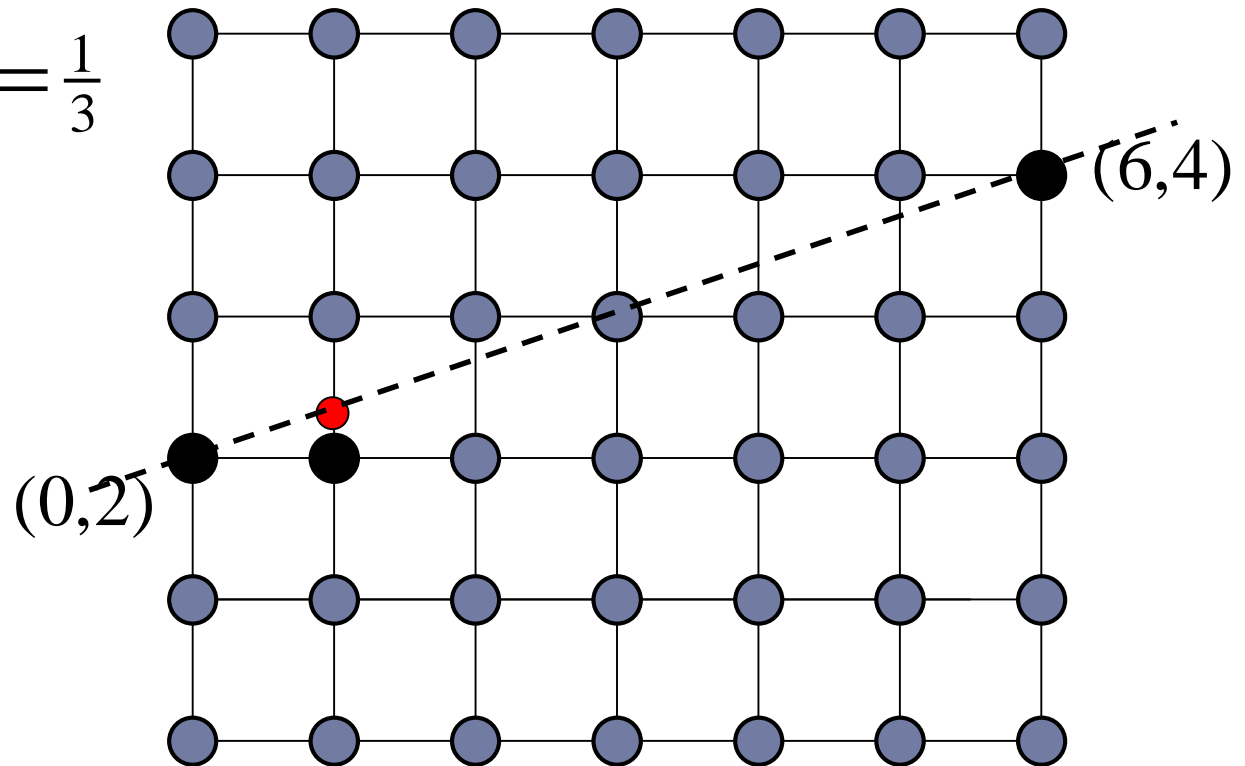
# DDA - Example

$$y = mx + b$$

$$m = \frac{y_H - y_L}{x_H - x_L} = \frac{4 - 2}{6 - 0} = \frac{1}{3}$$

$$x = 1$$

$$y = \frac{7}{3}$$



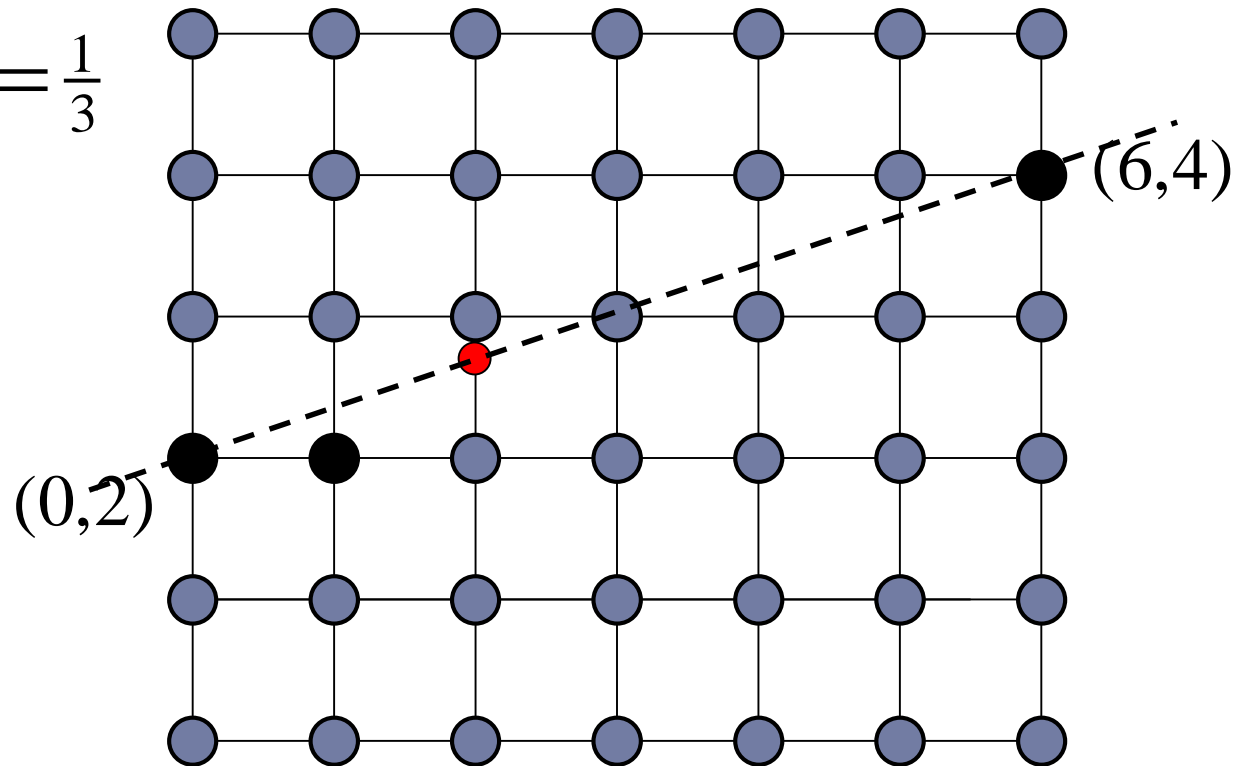
# DDA - Example

$$y = mx + b$$

$$m = \frac{y_H - y_L}{x_H - x_L} = \frac{4 - 2}{6 - 0} = \frac{1}{3}$$

$$x = 2$$

$$y = \frac{8}{3}$$



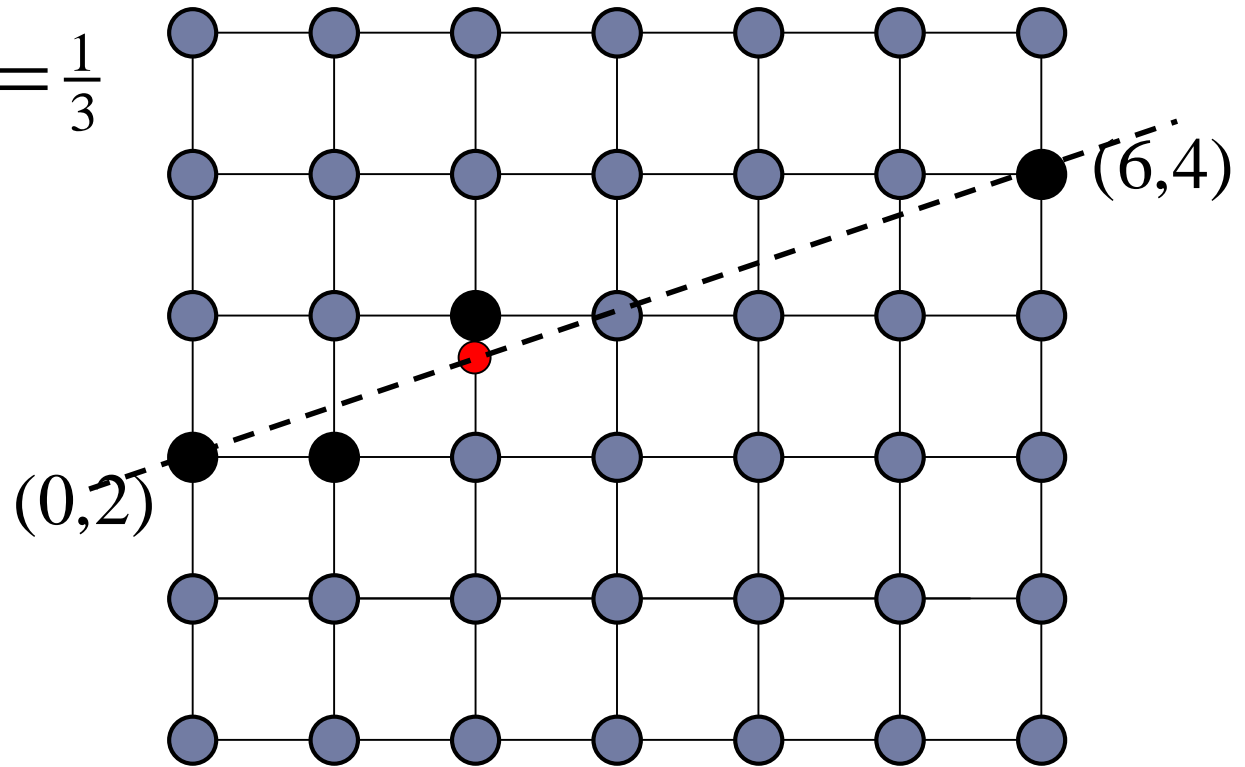
# DDA - Example

$$y = mx + b$$

$$m = \frac{y_H - y_L}{x_H - x_L} = \frac{4 - 2}{6 - 0} = \frac{1}{3}$$

$$x = 2$$

$$y = \frac{8}{3}$$



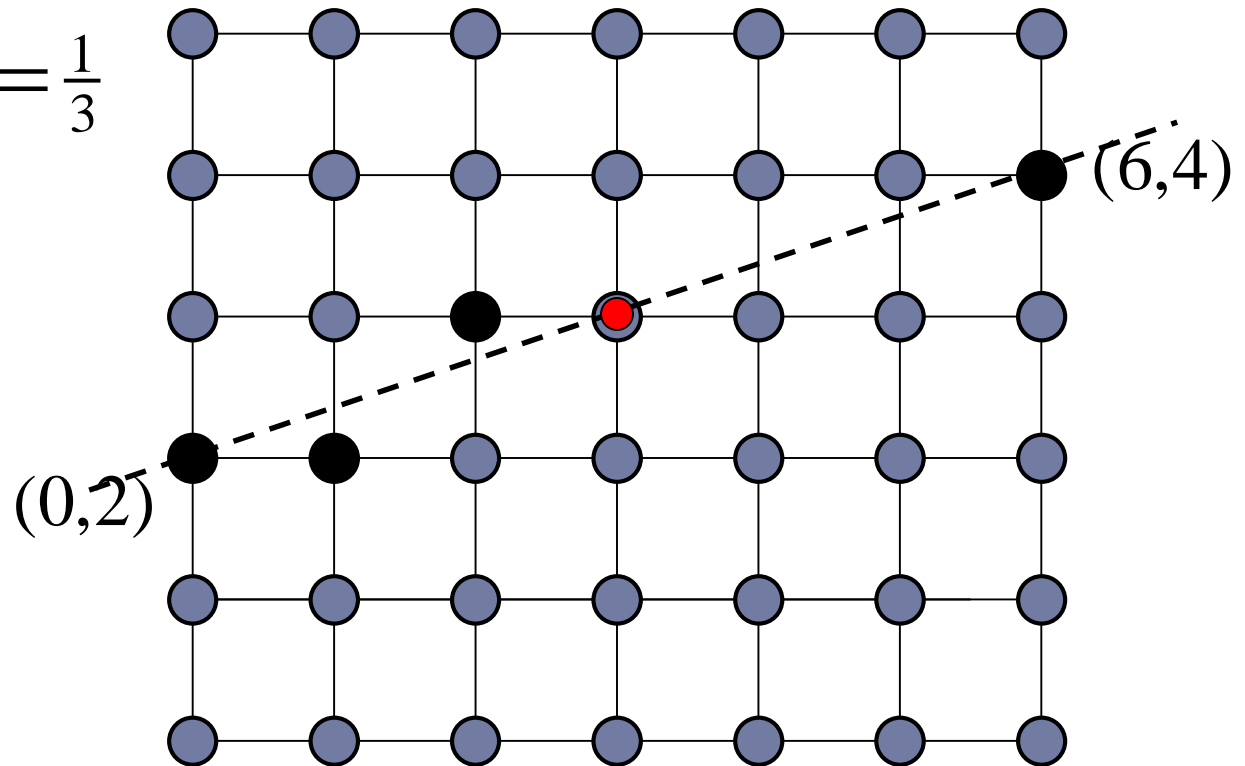
# DDA - Example

$$y = mx + b$$

$$m = \frac{y_H - y_L}{x_H - x_L} = \frac{4 - 2}{6 - 0} = \frac{1}{3}$$

$$x = 3$$

$$y = 3$$



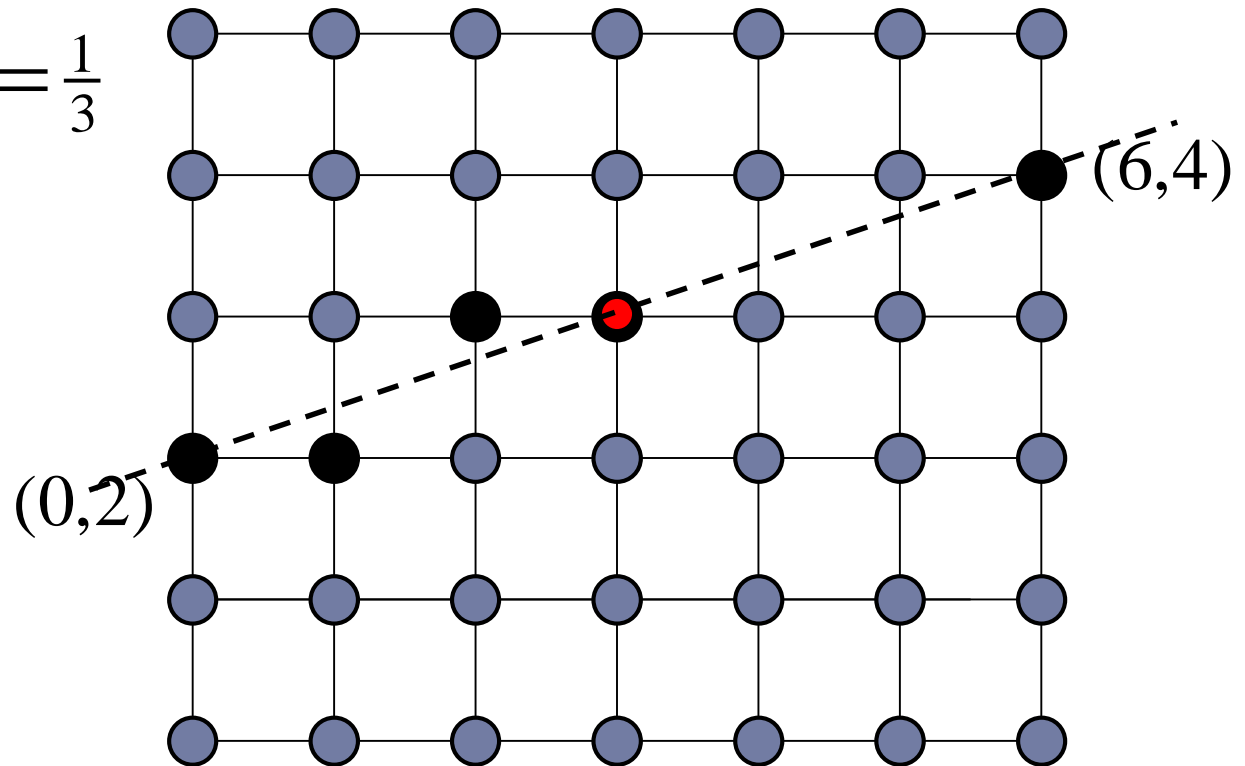
# DDA - Example

$$y = mx + b$$

$$m = \frac{y_H - y_L}{x_H - x_L} = \frac{4 - 2}{6 - 0} = \frac{1}{3}$$

$$x = 3$$

$$y = 3$$



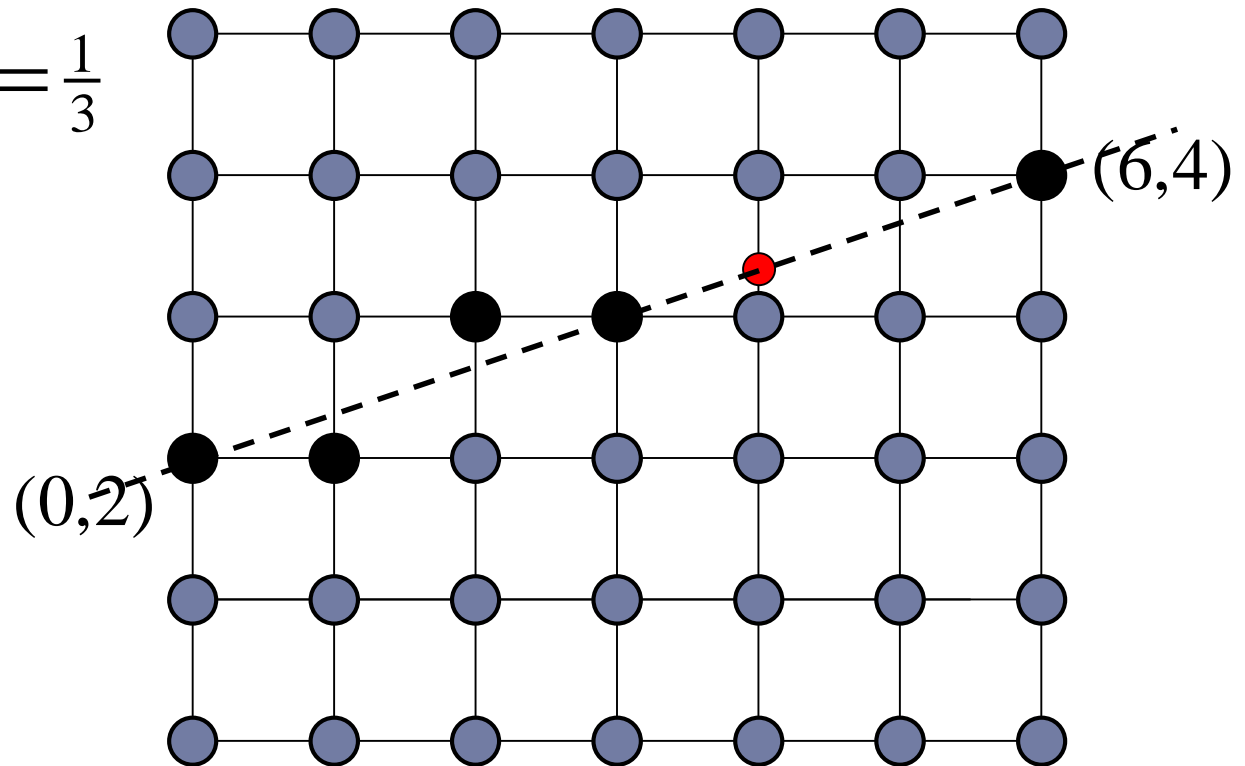
# DDA - Example

$$y = mx + b$$

$$m = \frac{y_H - y_L}{x_H - x_L} = \frac{4 - 2}{6 - 0} = \frac{1}{3}$$

$$x = 4$$

$$y = \frac{10}{3}$$



# DDA - Example

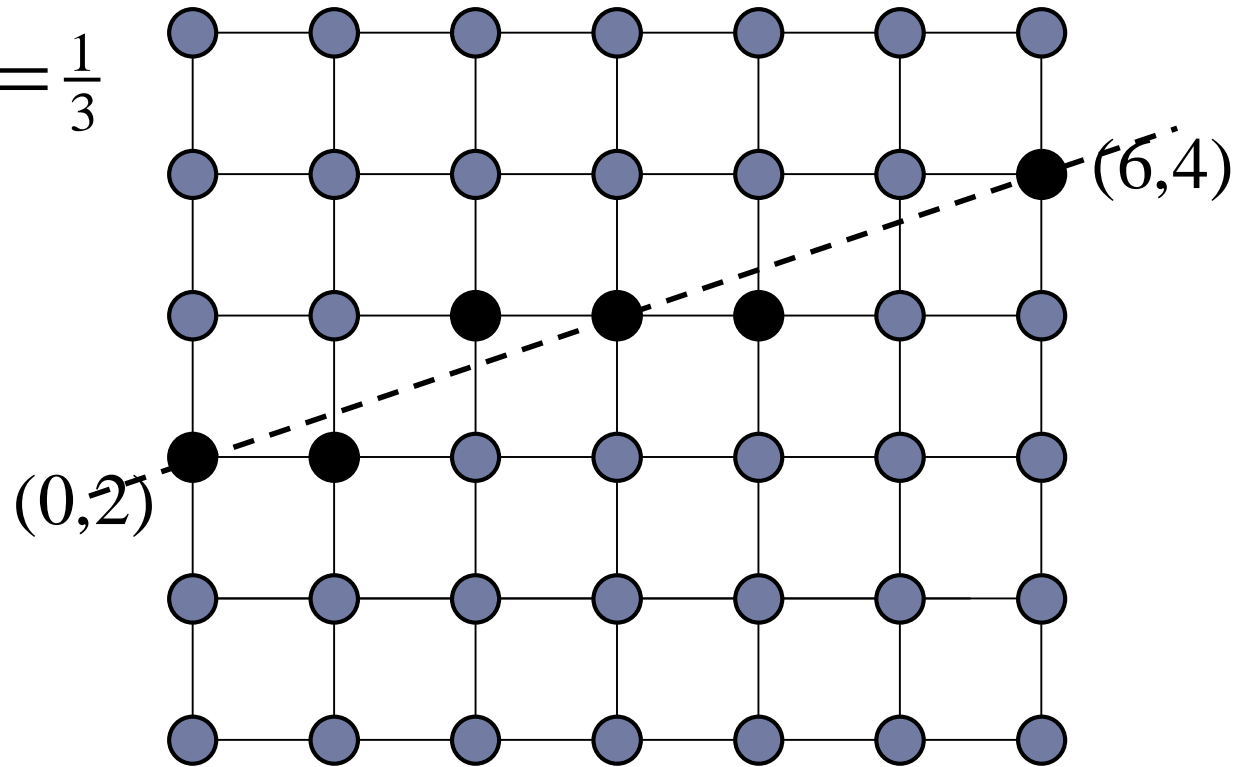
---

$$y = mx + b$$

$$m = \frac{y_H - y_L}{x_H - x_L} = \frac{4 - 2}{6 - 0} = \frac{1}{3}$$

$$x = 4$$

$$y = \frac{10}{3}$$



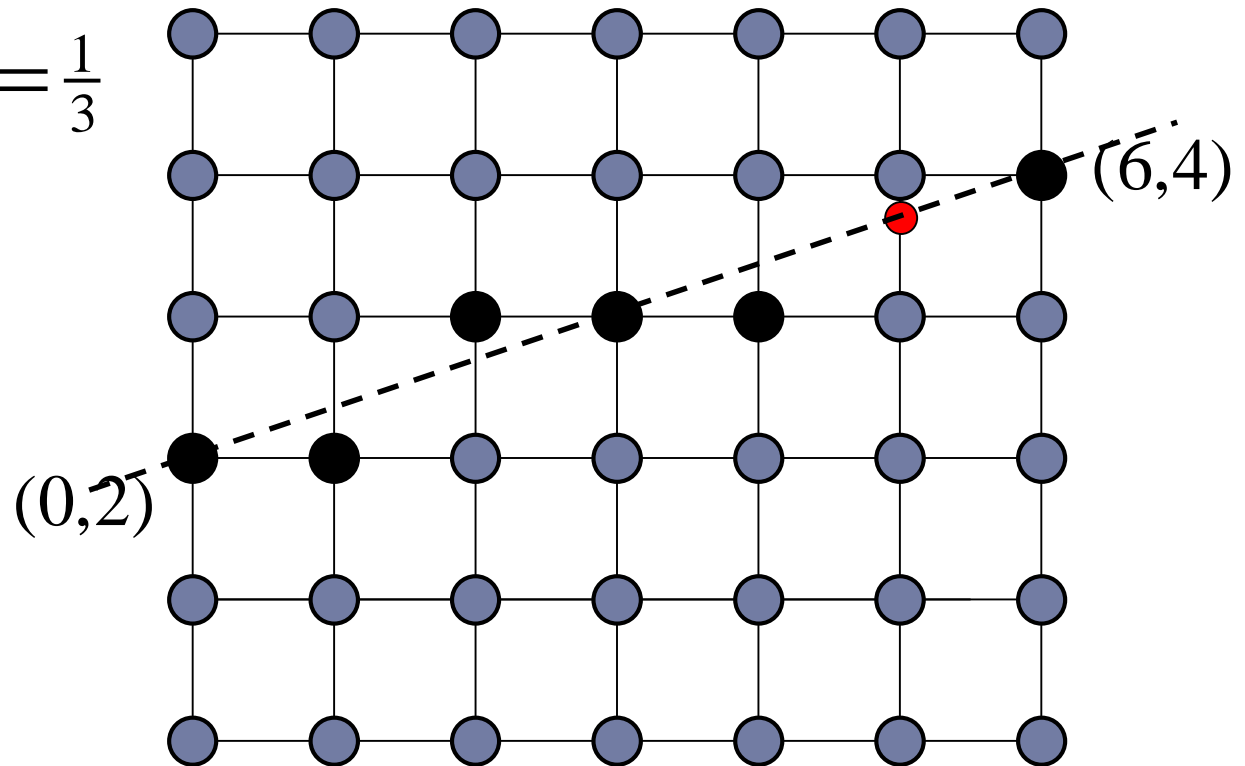
# DDA - Example

$$y = mx + b$$

$$m = \frac{y_H - y_L}{x_H - x_L} = \frac{4 - 2}{6 - 0} = \frac{1}{3}$$

$$x = 5$$

$$y = \frac{11}{3}$$





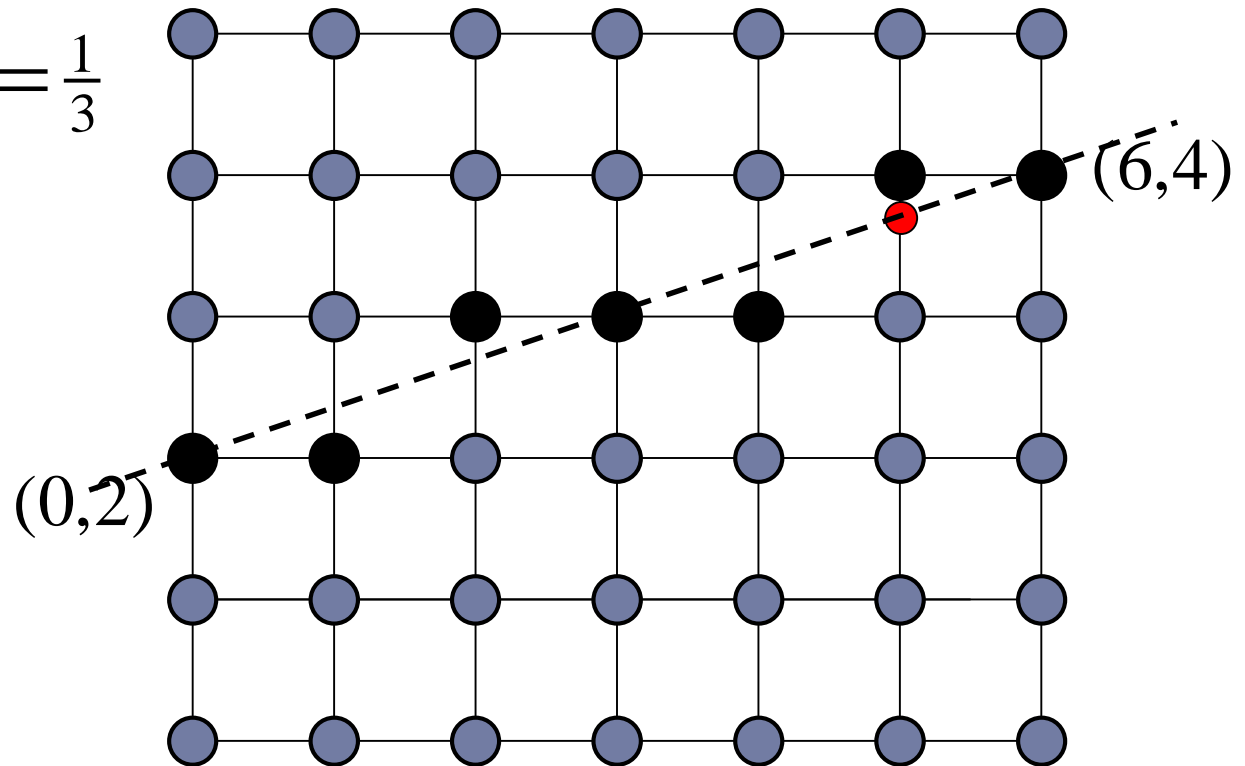
# DDA - Example

$$y = mx + b$$

$$m = \frac{y_H - y_L}{x_H - x_L} = \frac{4 - 2}{6 - 0} = \frac{1}{3}$$

$$x = 5$$

$$y = \frac{11}{3}$$



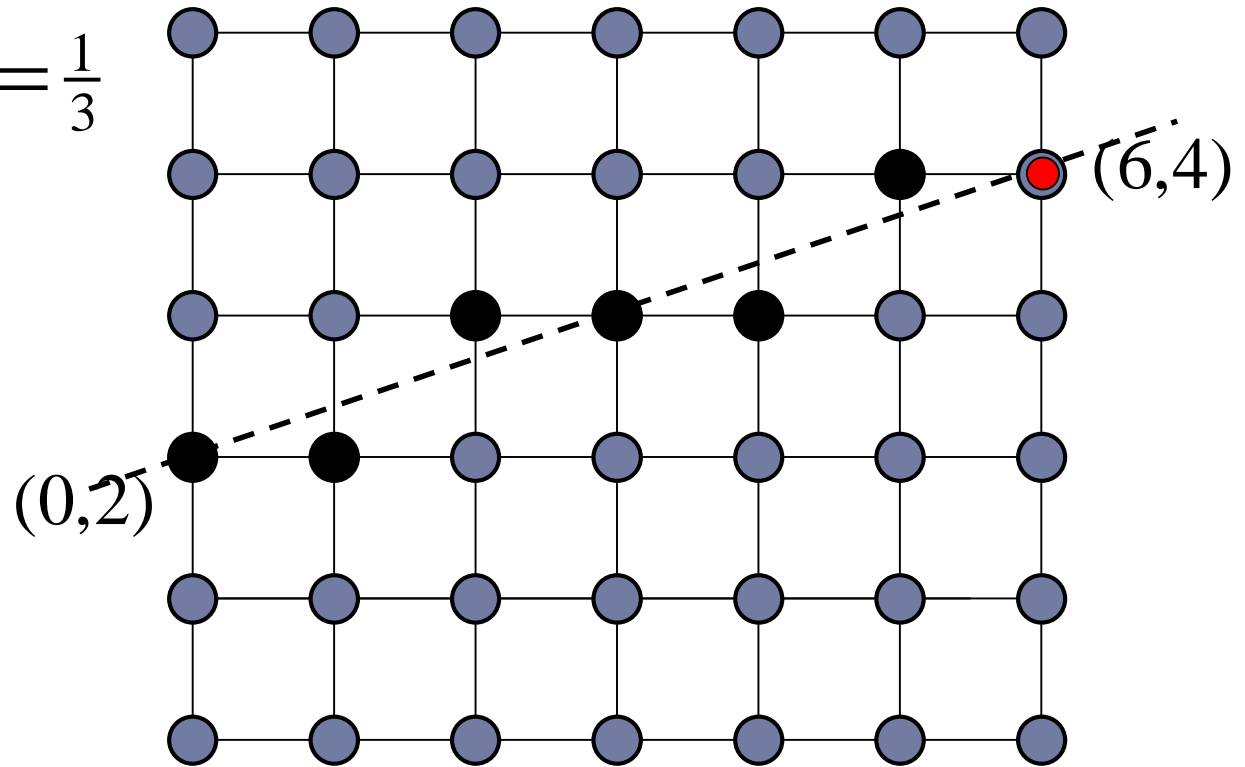
# DDA - Example

$$y = mx + b$$

$$m = \frac{y_H - y_L}{x_H - x_L} = \frac{4 - 2}{6 - 0} = \frac{1}{3}$$

$$x = 6$$

$$y = 4$$



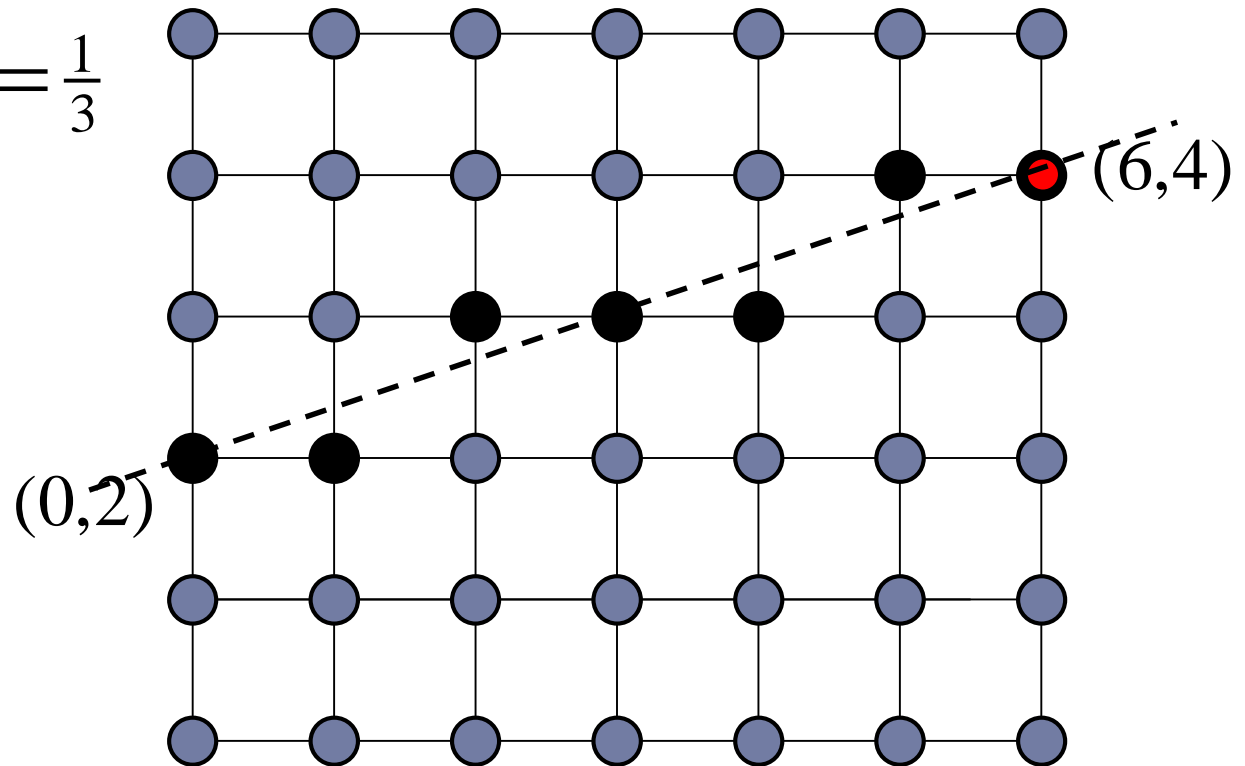
# DDA - Example

$$y = mx + b$$

$$m = \frac{y_H - y_L}{x_H - x_L} = \frac{4 - 2}{6 - 0} = \frac{1}{3}$$

$$x = 6$$

$$y = 4$$



# DDA - Γενικός - Άσκηση

---

$dx = x_H - x_L$

$dy = y_H - y_L$

$(x_0, y_0) = (x_L, y_L)$

if  $|dx| > |dy|$

    step = dx

else

    step = dy

$x_{inc} = dx / step$

$y_{inc} = dy / step$

for (  $i = 0; i \leq step; i++$  )

    DrawPixel ( round(  $x_i$  ), round (  $y_i$  ) )

$x_{i+1} = x_i + x_{inc}$

$y_{i+1} = y_i + y_{inc}$

# DDA - Προβλήματα

---

- ▶ Πράξεις κινητής υποδιαστολής
- ▶ Στρογγύλευση
- ▶ Βελτίωση;

$(x_0, y_0) = (x_L, y_L)$

For (  $i = 0$ ;  $i \leq x_H - x_L$ ;  $i++$  )

    DrawPixel (  $x_i$ , Round (  $y_i$  ) )

$x_{i+1} = x_i + 1$

$y_{i+1} = y_i + m$