

ΗΥ416 ΓΡΑΦΙΚΑ ΥΠΟΛΟΓΙΣΤΩΝ

Σχεδίαση στον ΗΥ

Π. ΤΣΟΜΠΑΝΟΠΟΥΛΟΥ

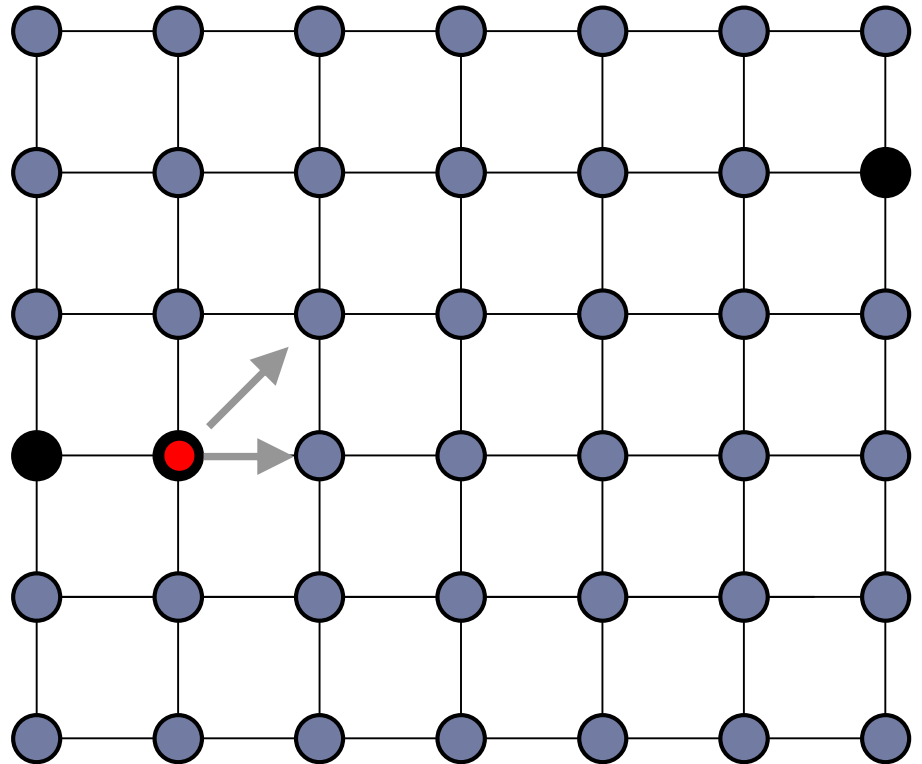
ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ & ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

Midpoint Algorithm (Αλγόριθμος Μέσου Σημείου)

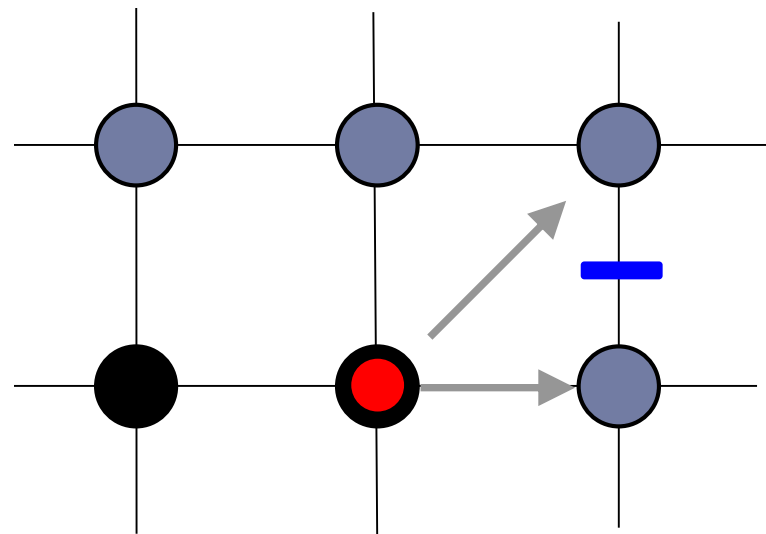
Midpoint Algorithm

- ▶ Given a point just drawn, determine whether we move E or NE on next step

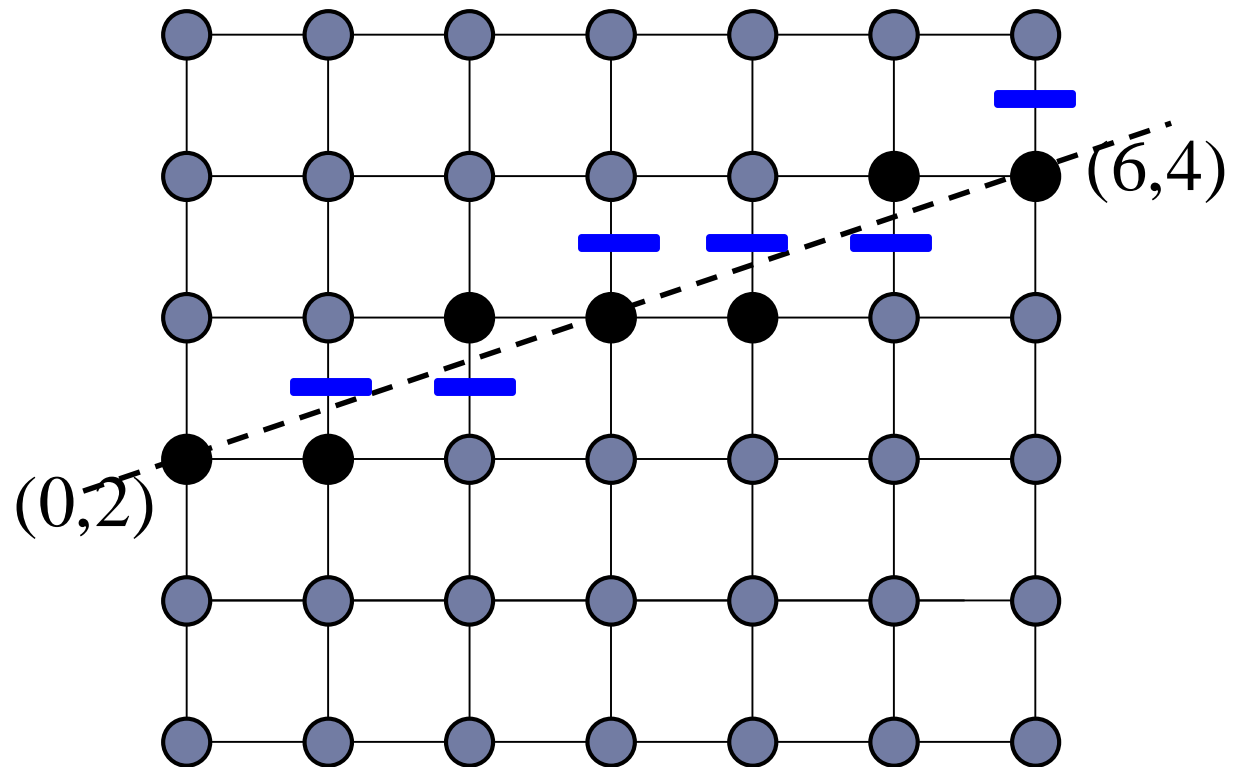


Midpoint Algorithm

- ▶ Given a point just drawn, determine whether we move E or NE on next step
- ▶ Is the line above or below $(x+1, y + \frac{1}{2})$?
Below: move E
Above: move NE



Above/Below the Line?

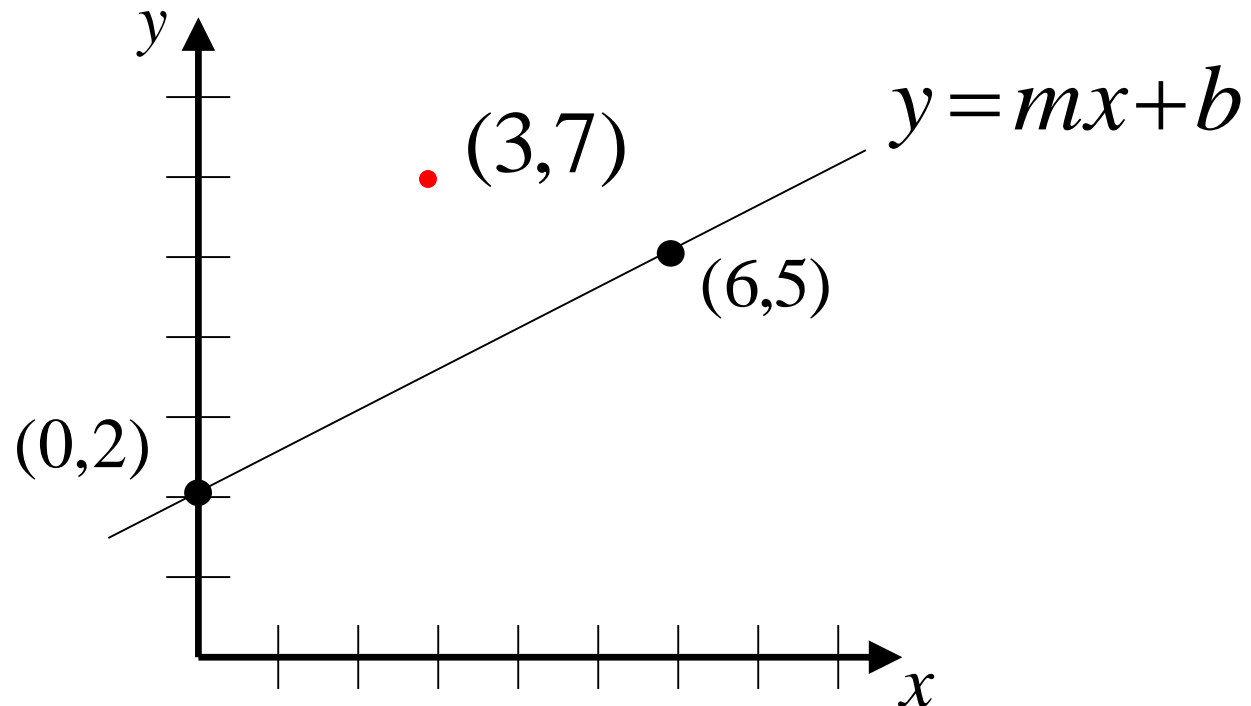


Δύο προβλήματα

- ▶ Πώς αξιολογείται αν το μέσο είναι πάνω ή κάτω από τη γραμμή;
- ▶ Πώς να αξιολογηθεί σταδιακά αυτό;

Midpoint Algorithm - Έμμεση μορφή γραμμής

- ▶ Πώς καταλαβαίνετε εάν ένα σημείο είναι πάνω ή κάτω από τη γραμμή;



Midpoint Algorithm - Έμμεση μορφή γραμμής

- ▶ Πώς καταλαβαίνετε εάν ένα σημείο είναι πάνω ή κάτω από τη γραμμή;

$$y = mx + b$$

Midpoint Algorithm - Έμμεση μορφή γραμμής

- ▶ Πώς καταλαβαίνετε εάν ένα σημείο είναι πάνω ή κάτω από τη γραμμή;

$$y = \frac{y_H - y_L}{x_H - x_L} x + b$$

Midpoint Algorithm - Έμμεση μορφή γραμμής

- ▶ Πώς καταλαβαίνετε εάν ένα σημείο είναι πάνω ή κάτω από τη γραμμή;

$$(x_H - x_L)y = \left(\frac{y_H - y_L}{x_H - x_L}x + b\right)(x_H - x_L)$$

Midpoint Algorithm - Έμμεση μορφή γραμμής

- ▶ Πώς καταλαβαίνετε εάν ένα σημείο είναι πάνω ή κάτω από τη γραμμή;

$$(x_H - x_L)y = (y_H - y_L)x + (x_H - x_L)b$$

Midpoint Algorithm - Έμμεση μορφή γραμμής

- ▶ Πώς καταλαβαίνετε εάν ένα σημείο είναι πάνω ή κάτω από τη γραμμή;

$$(x_H - x_L)y + (y_L - y_H)x + (x_L - x_H)b = 0$$

Midpoint Algorithm - Έμμεση μορφή γραμμής

- ▶ Πώς καταλαβαίνετε εάν ένα σημείο είναι πάνω ή κάτω από τη γραμμή;

$$(x_H - x_L)y + (y_L - y_H)x + (x_L - x_H)b = 0$$

$$f(x, y) = cx + dy + e$$

$$c = y_L - y_H \quad d = x_H - x_L \quad e = b(x_L - x_H)$$

Midpoint Algorithm - Έμμεση μορφή γραμμής

- Πώς καταλαβαίνετε εάν ένα σημείο είναι πάνω ή κάτω από τη γραμμή;

$$(x_H - x_L)y + (y_L - y_H)x + (x_L - x_H)b = 0$$

$$f(x, y) = cx + dy + e$$

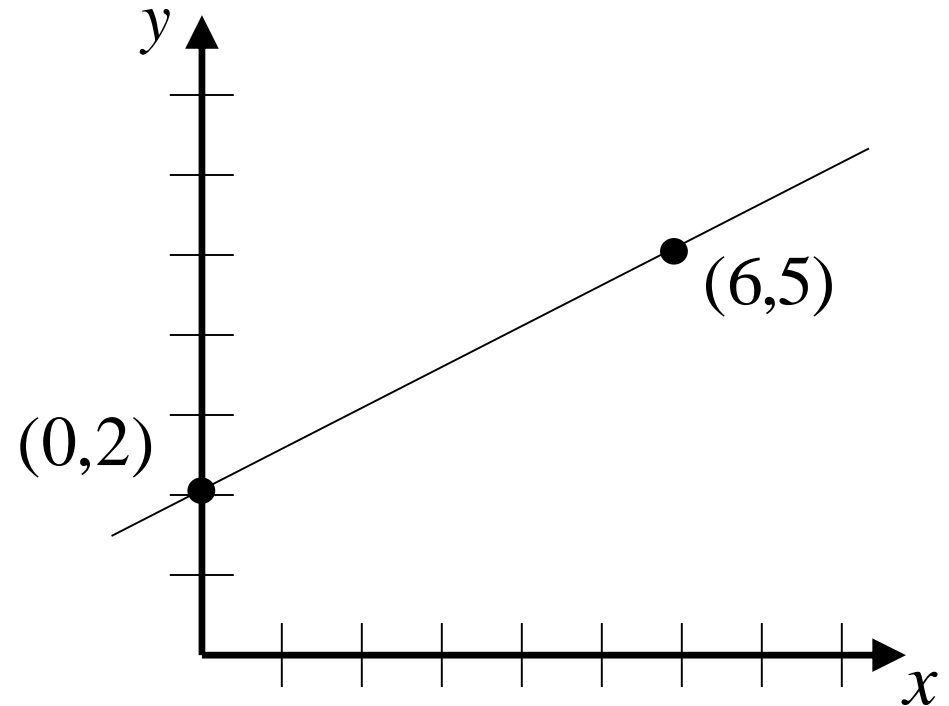
$$c = y_L - y_H \quad d = x_H - x_L \quad e = b(x_L - x_H)$$

Properties

$f(x, y) = 0$	(x, y) on the line
$f(x, y) < 0$	(x, y) below the line
$f(x, y) > 0$	(x, y) above the line

Midpoint Algorithm - Έμμεση μορφή γραμμής

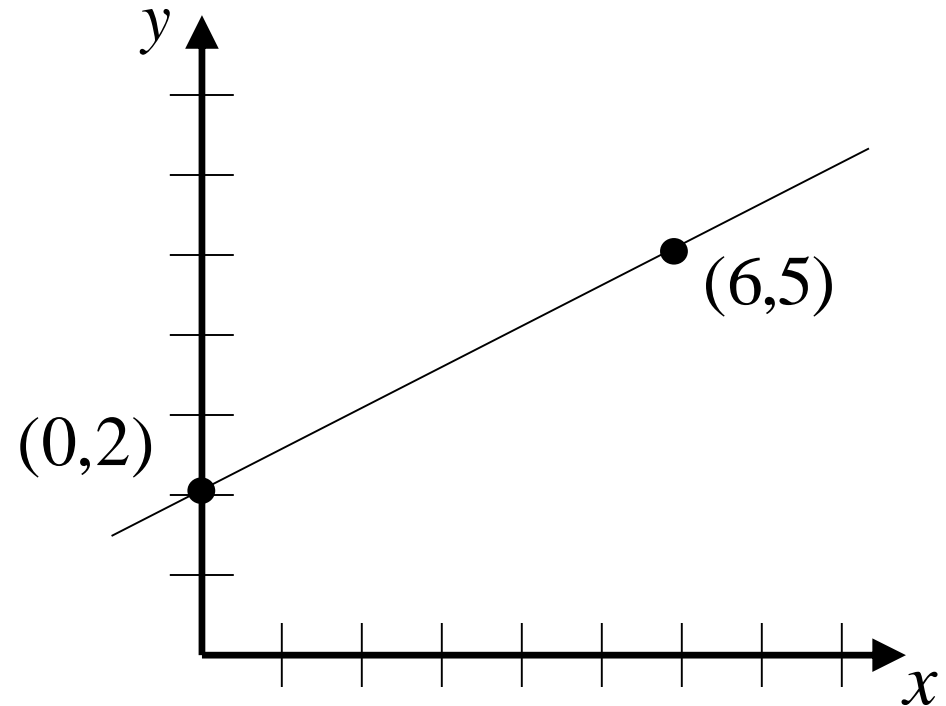
$$y = \frac{3}{6}x + 2$$



Midpoint Algorithm - Implicit Forms

$$y = \frac{3}{6}x + 2$$

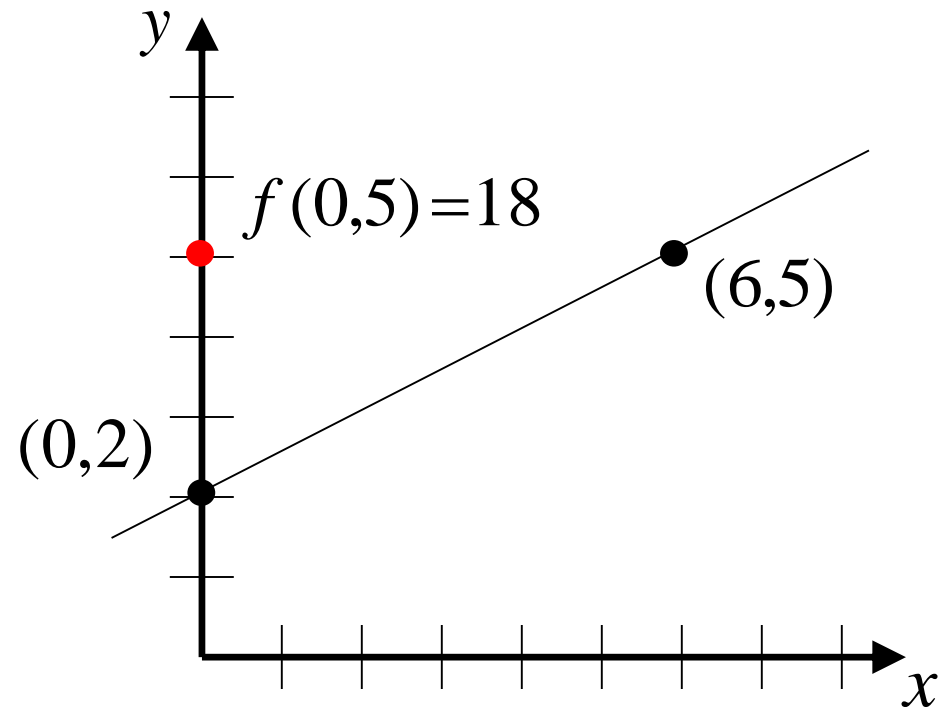
$$f(x, y) = 6y - 3x - 12$$



Midpoint Algorithm - Implicit Forms

$$y = \frac{3}{6}x + 2$$

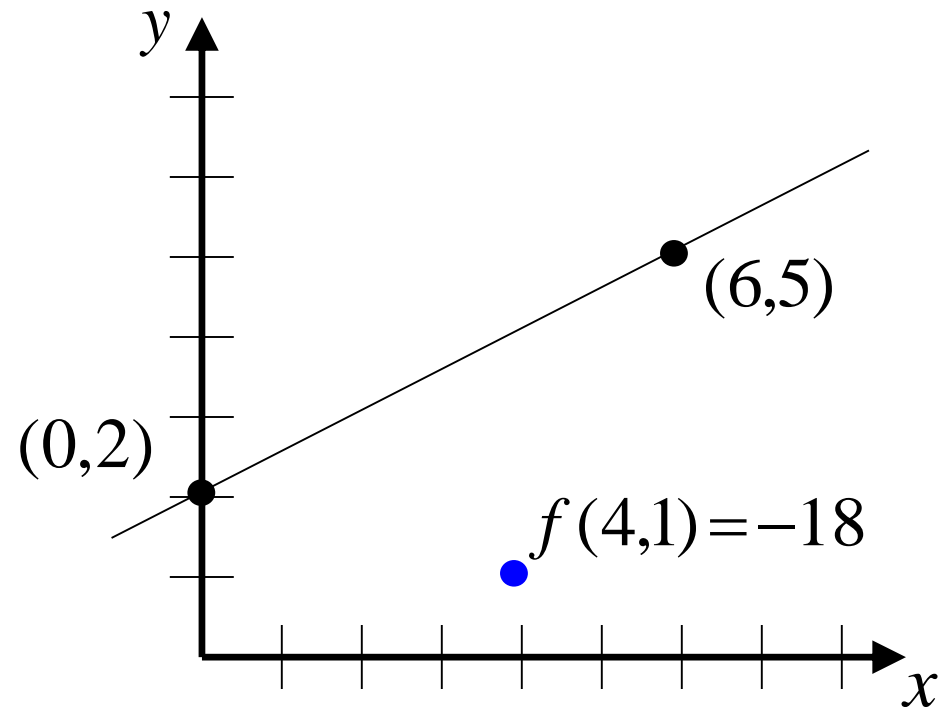
$$f(x, y) = 6y - 3x - 12$$



Midpoint Algorithm - Implicit Forms

$$y = \frac{3}{6}x + 2$$

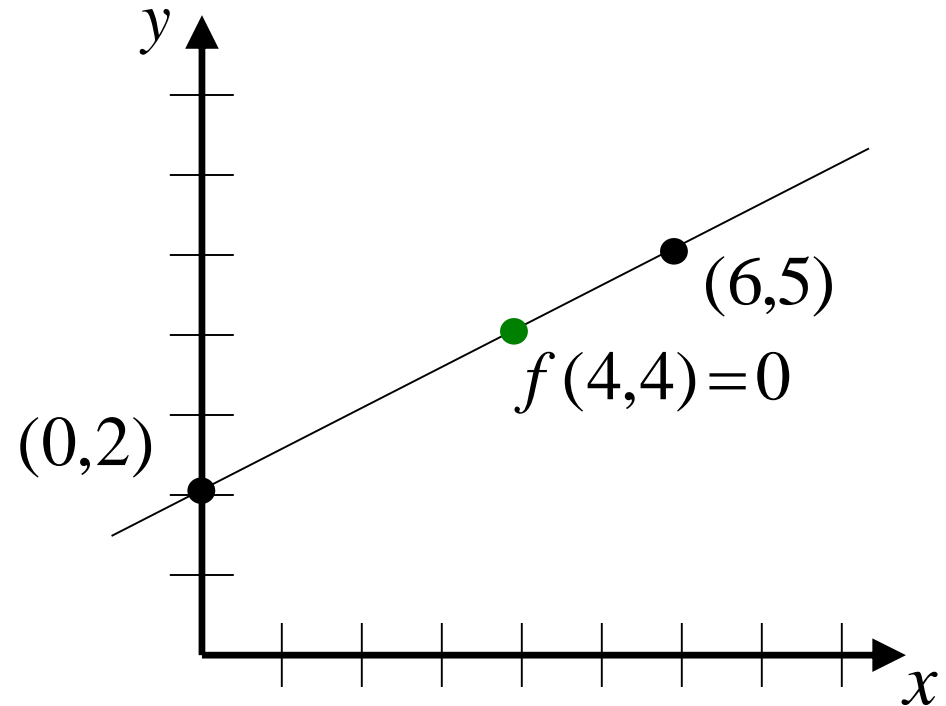
$$f(x, y) = 6y - 3x - 12$$



Midpoint Algorithm - Implicit Forms

$$y = \frac{3}{6}x + 2$$

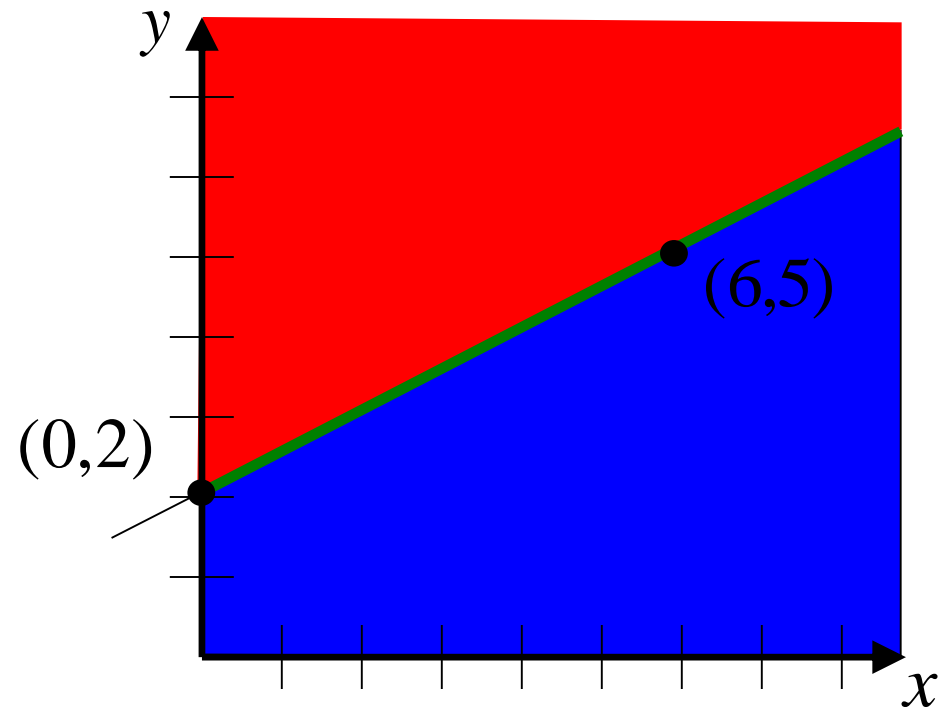
$$f(x, y) = 6y - 3x - 12$$



Midpoint Algorithm - Implicit Forms

$$y = \frac{3}{6}x + 2$$

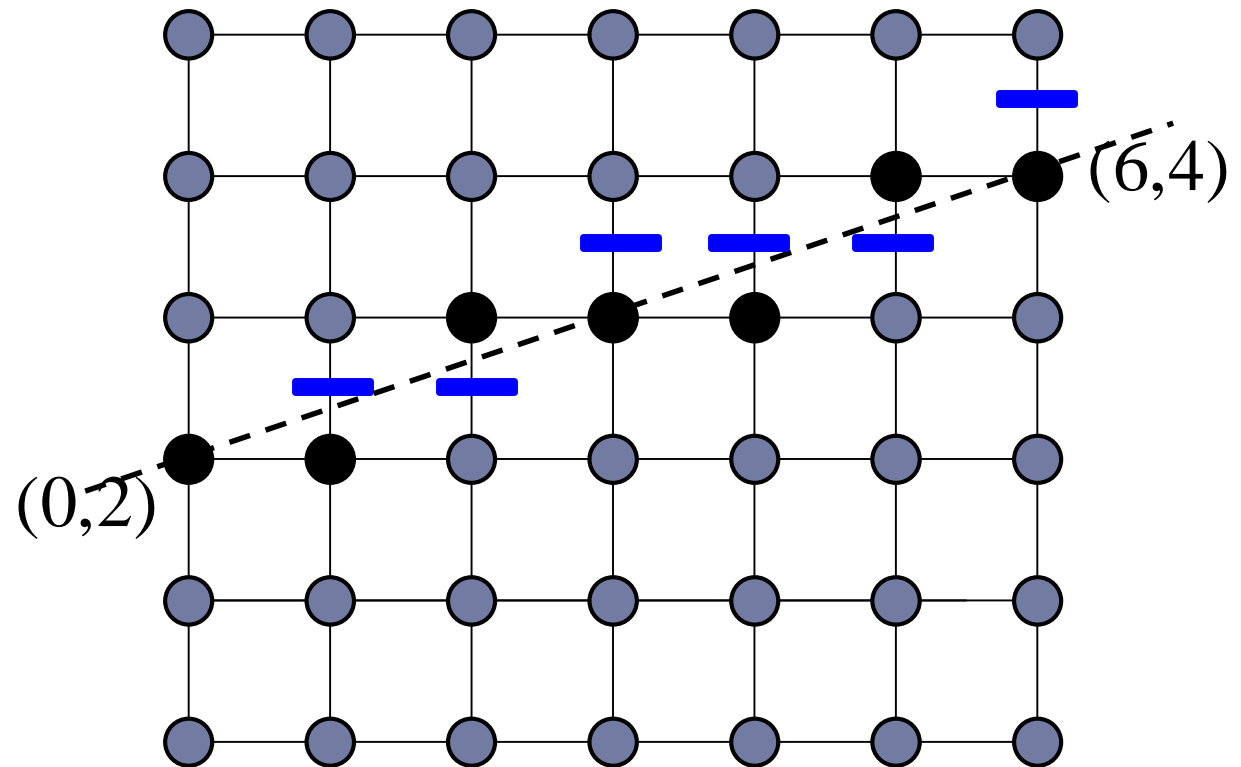
$$f(x, y) = 6y - 3x - 12$$



Δύο προβλήματα

- ▶ Πώς αξιολογείται αν το μέσο είναι πάνω ή κάτω από τη γραμμή;
- ▶ Πώς να αξιολογηθεί σταδιακά αυτό;

Πάνω/Κάτω από τη γραμμή?



Midpoint Algorithm

What about starting value?

What is the middle point?

Midpoint Algorithm

What about starting value?

What is the middle point? $(x_{L+1}, y_{L+1/2})$

Midpoint Algorithm

What about starting value?

What is the middle point? $(x_{L+1}, y_{L+1/2})$

$$f(x_L + 1, y_L + \frac{1}{2}) = c(x_L + 1) + d(y_L + \frac{1}{2}) + e$$

Midpoint Algorithm

What about starting value?

$$f(x_L + 1, y_L + \frac{1}{2}) = c(x_L + 1) + d(y_L + \frac{1}{2}) + e$$

$$f(x_L + 1, y_L + \frac{1}{2}) = f(x_L, y_L) + c + \frac{1}{2}d$$

Midpoint Algorithm

What about starting value?

$$f(x_L + 1, y_L + \frac{1}{2}) = c(x_L + 1) + d(y_L + \frac{1}{2}) + e$$

$$f(x_L + 1, y_L + \frac{1}{2}) = \cancel{f(x_L, y_L)} + c + \frac{1}{2}d$$

(x_L, y_L) is on the
line!

Midpoint Algorithm

What about starting value?

$$f(x_L + 1, y_L + \frac{1}{2}) = c(x_L + 1) + d(y_L + \frac{1}{2}) + e$$

$$f(x_L + 1, y_L + \frac{1}{2}) = c + \frac{1}{2}d$$

Midpoint Algorithm

What about starting value?

$$f(x_L + 1, y_L + \frac{1}{2}) = 2c(x_L + 1) + 2d(y_L + \frac{1}{2}) + 2e$$

$$f(x_L + 1, y_L + \frac{1}{2}) = 2c + d$$

Πολλαπλασιάζοντας με 2 τους συντελεστές δεν αλλάζει το πρόσημο της f

Midpoint Algorithm

- ▶ Χρειάζεται η τιμή στο $f(x+1, y+\frac{1}{2})$ για να καθορίσεις πως θα προχωρήσεις (E or NE)
- ▶ Αλγόριθμος σταδιακής κατασκευής
- ▶ Έστω ότι γνωρίζουμε την $f(x+1, y+\frac{1}{2})$
 - ▶ Υπολόγισε την $f(x+2, y+\frac{1}{2})$, αν έχει επιλεγεί E
γιατί επιλέχθηκε/χρωματίστηκε το σημείο $(x+1, y)$
 - ▶ Υπολόγισε την $f(x+2, y+\frac{3}{2})$, αν έχει επιλεγεί NE
γιατί επιλέχθηκε/χρωματίστηκε το σημείο $(x+1, y+1)$

Midpoint Algorithm

Αν έχει επιλεγεί E (δηλ. $(x+1, y)$), υπολόγισε

$$f(x+2, y+\frac{1}{2})$$

$$f(x+2, y+\frac{1}{2}) = 2c(x+2) + 2d(y+\frac{1}{2}) + 2e$$

Midpoint Algorithm

Αν έχει επιλεγεί E (δηλ. $(x+1, y)$), υπολόγισε

$$f(x+2, y+\frac{1}{2})$$

$$f(x+2, y+\frac{1}{2}) = 2c(x+2) + 2d(y+\frac{1}{2}) + 2e$$

$$f(x+2, y+\frac{1}{2}) = 2c + f(x+1, y+\frac{1}{2})$$

Midpoint Algorithm

Αν έχει επιλεγεί ΝΕ (δηλ. $(x+1, y+1)$), υπολόγισε

$$f(x+2, y+\frac{3}{2})$$

$$f(x+2, y+\frac{3}{2}) = 2c(x+2) + 2d(y+\frac{3}{2}) + 2e$$

Midpoint Algorithm

Αν έχει επιλεγεί ΝΕ (δηλ. $(x+1, y+1)$), υπολόγισε

$$f(x+2, y+\frac{3}{2})$$

$$f(x+2, y+\frac{3}{2}) = 2c(x+2) + 2d(y+\frac{3}{2}) + 2e$$

$$f(x+2, y+\frac{3}{2}) = 2c + 2d + f(x+1, y+\frac{1}{2})$$

Midpoint Algorithm - Σύνοψη

```
x=xL
y=yL
d=xH - xL
c=yL - yH
sum=2c+d
draw(x,y)
while ( x < xH )
    if ( sum < 0 )
        sum += 2d
        y++
    x++
    sum += 2c
    draw(x,y)
```

Midpoint Algorithm - Σύνοψη

$x = x_L$

$y = y_L$

$d = x_H - x_L$

$c = y_L - y_H$

$sum = 2c + d$ // αρχική τιμή - η f στο επόμενο σημείο

$draw(x, y)$

while ($x < x_H$)

 if ($sum < 0$)

$sum += 2d$

$y++$

$x++$

$sum += 2c$

$draw(x, y)$

Midpoint Algorithm - Σύνοψη

```
x=xL
y=yL
d=xH - xL
c=yL - yH
sum=2c+d // αρχική τιμή - η f στο επόμενο σημείο
draw(x,y)
while ( x < xH) // για όλα τα σημεία
    if ( sum < 0 ) // ενδιαμέσο σημείο κάτω από τη γραμμή → επέλεξε το (x+1, y+1)
        sum += 2d
        y++
    x++
    sum += 2c
    draw(x,y)
```

Midpoint Algorithm - Σύνοψη

```
x=xL
y=yL
d=xH - xL
c=yL - yH
sum=2c+d // αρχική τιμή - η f στο επόμενο σημείο
draw(x,y)
while ( x < xH) // για όλα τα σημεία
    if ( sum < 0 ) // ενδιαμέσο σημείο κάτω από τη γραμμή → επέλεξε το (x+1, y+1)
        sum += 2d // η f στο επόμενο σημείο
        y++
    x++
    sum += 2c // η f στο επόμενο σημείο
    draw(x,y)
```

Midpoint Algorithm - Παράδειγμα

$x = x_L$

$y = y_L$

$d = x_H - x_L$

$c = y_L - y_H$

$sum = 2c + d$ ←

$draw(x, y)$

while ($x < x_H$)

 if ($sum < 0$)

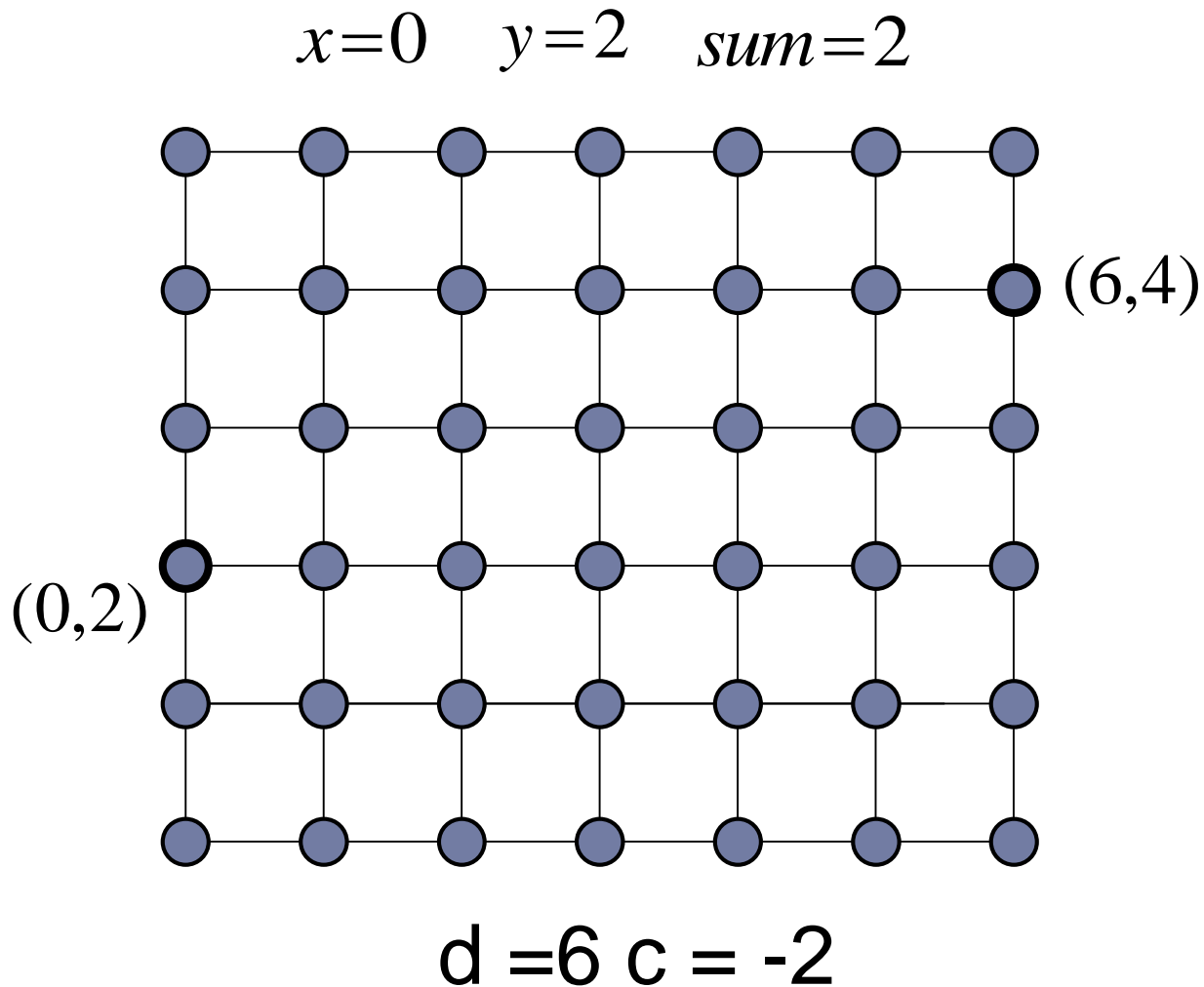
$sum += 2d$

$y++$

$x++$

$sum += 2c$

$draw(x, y)$



Midpoint Algorithm - Παράδειγμα

$x = x_L$

$y = y_L$

$d = x_H - x_L$

$c = y_L - y_H$

$sum = 2c + d$

$draw(x, y)$

$while (x < x_H)$ ←

if ($sum < 0$)

$sum += 2d$

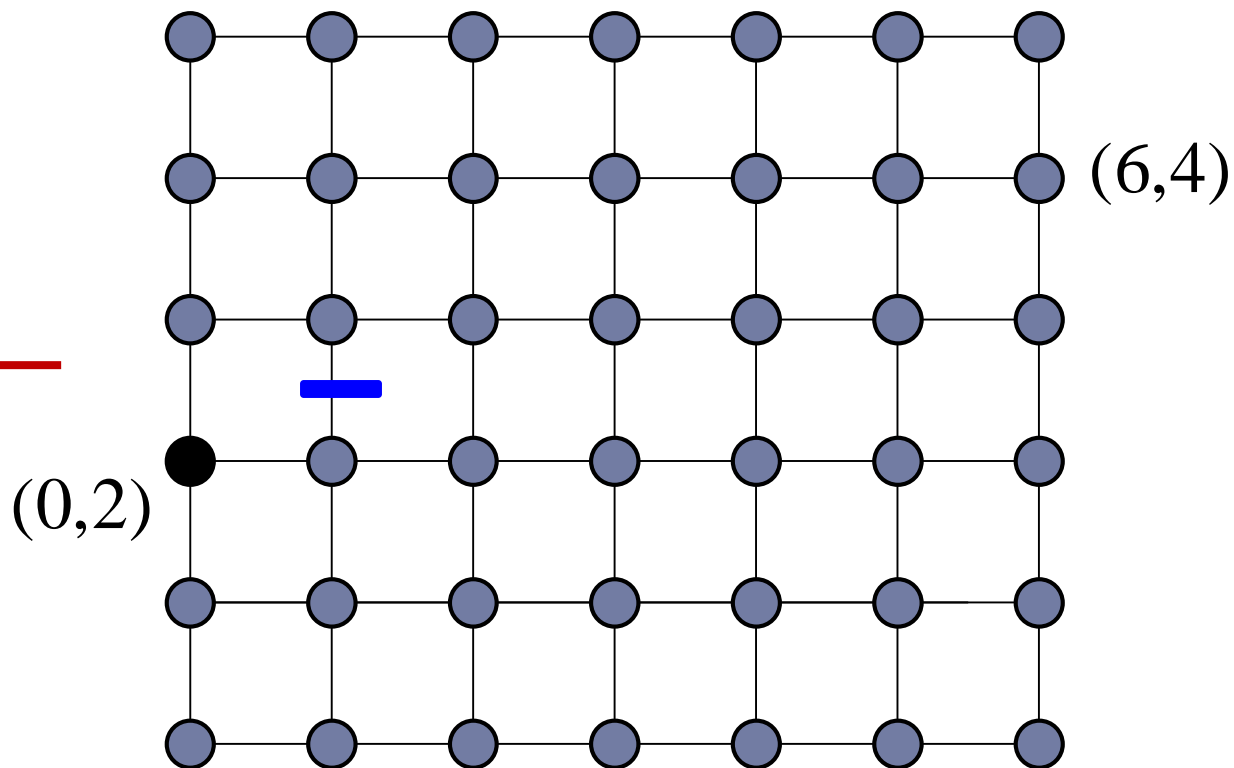
$y++$

$x++$

$sum += 2c$

$draw(x, y)$

$x = 0 \quad y = 2 \quad sum = 2$



$d = 6 \quad c = -2$

Midpoint Algorithm - Παράδειγμα

$x = x_L$

$y = y_L$

$d = x_H - x_L$

$c = y_L - y_H$

$sum = 2c + d$

$draw(x, y)$

while ($x < x_H$)

 if ($sum < 0$)

$sum += 2d$

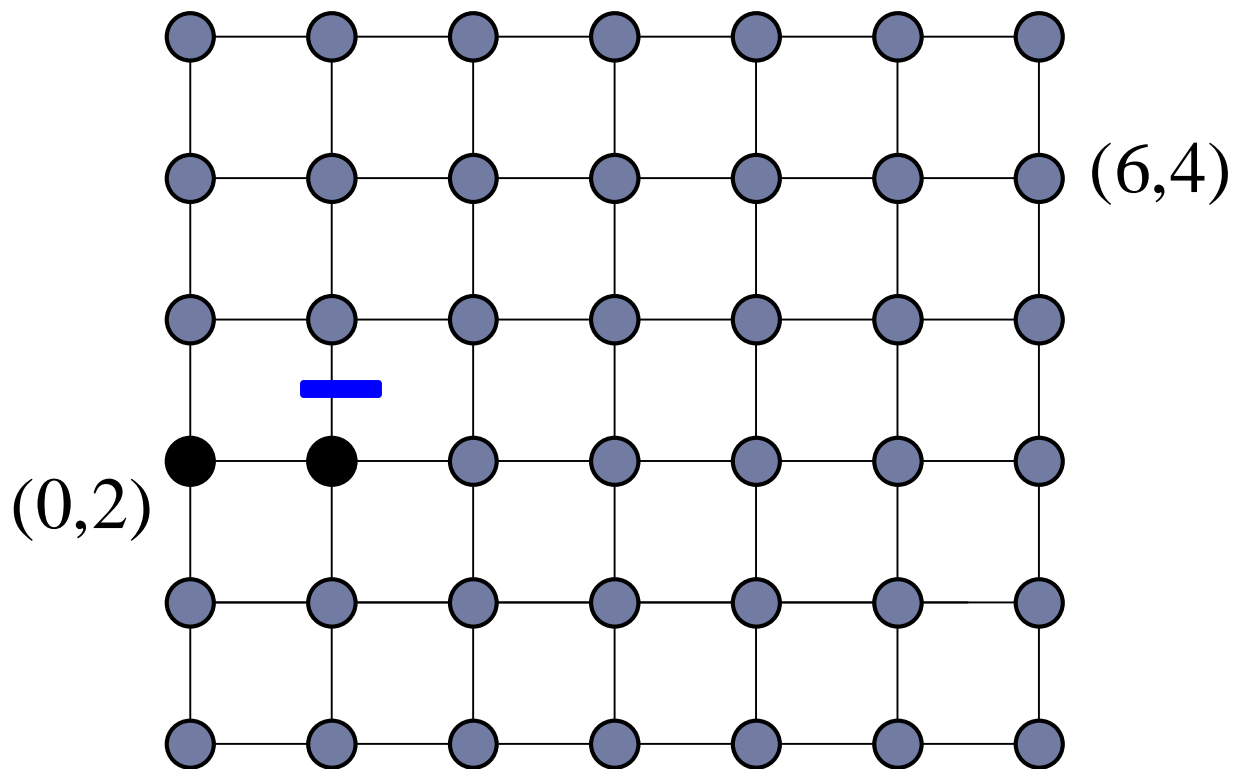
$y++$

$x++$

$sum += 2c$

$draw(x, y)$ ←

$x = 1$ $y = 2$ $sum = -2$



$d = 6$ $c = -2$

Midpoint Algorithm - Παράδειγμα

$x = x_L$

$y = y_L$

$d = x_H - x_L$

$c = y_L - y_H$

$sum = 2c + d$

$draw(x, y)$

while ($x < x_H$)

if ($sum < 0$)

$sum += 2d$

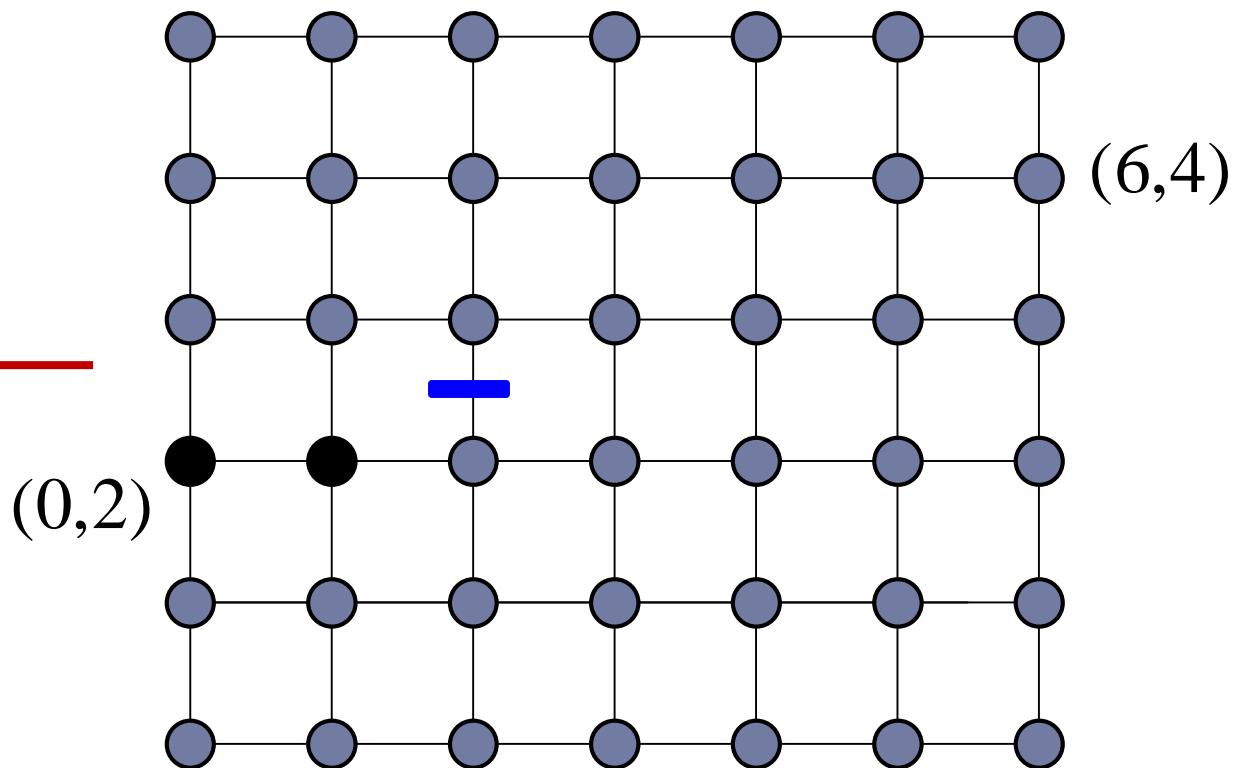
$y++$

$x++$

$sum += 2c$

$draw(x, y)$

$x=1 \quad y=2 \quad sum=-2$



$d = 6 \quad c = -2$

Midpoint Algorithm - Παράδειγμα

$x = x_L$

$y = y_L$

$d = x_H - x_L$

$c = y_L - y_H$

$sum = 2c + d$

$draw(x, y)$

while ($x < x_H$)

if ($sum < 0$)

$sum += 2d$

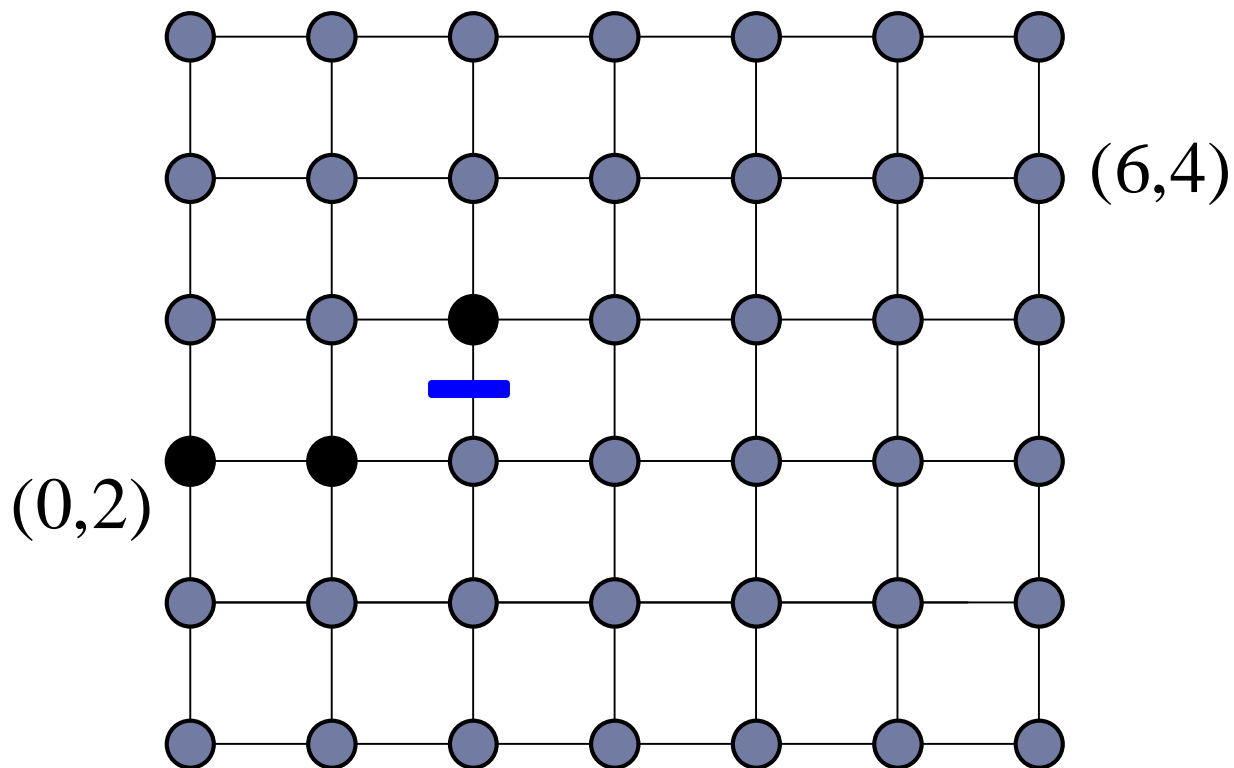
$y++$

$x++$

$sum += 2c$

$draw(x, y)$ ←

$x = 2 \quad y = 3 \quad sum = 6$



$d = 6 \quad c = -2$

Midpoint Algorithm - Παράδειγμα

$x = x_L$

$y = y_L$

$d = x_H - x_L$

$c = y_L - y_H$

$sum = 2c + d$

$draw(x, y)$

while ($x < x_H$)

if ($sum < 0$)

$sum += 2d$

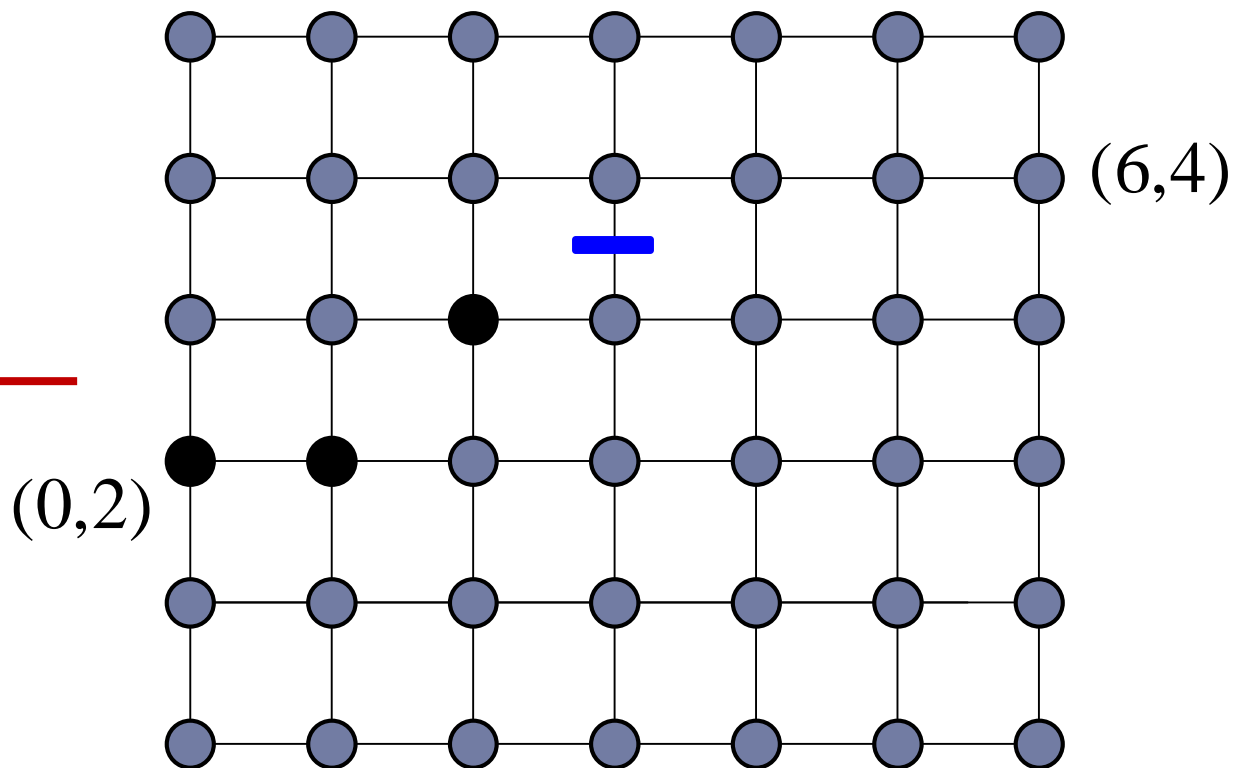
$y++$

$x++$

$sum += 2c$

$draw(x, y)$

$x=2$ $y=3$ $sum=6$



Midpoint Algorithm - Παράδειγμα

$x = x_L$

$y = y_L$

$d = x_H - x_L$

$c = y_L - y_H$

$sum = 2c + d$

$draw(x, y)$

while ($x < x_H$)

if ($sum < 0$)

$sum += 2d$

$y++$

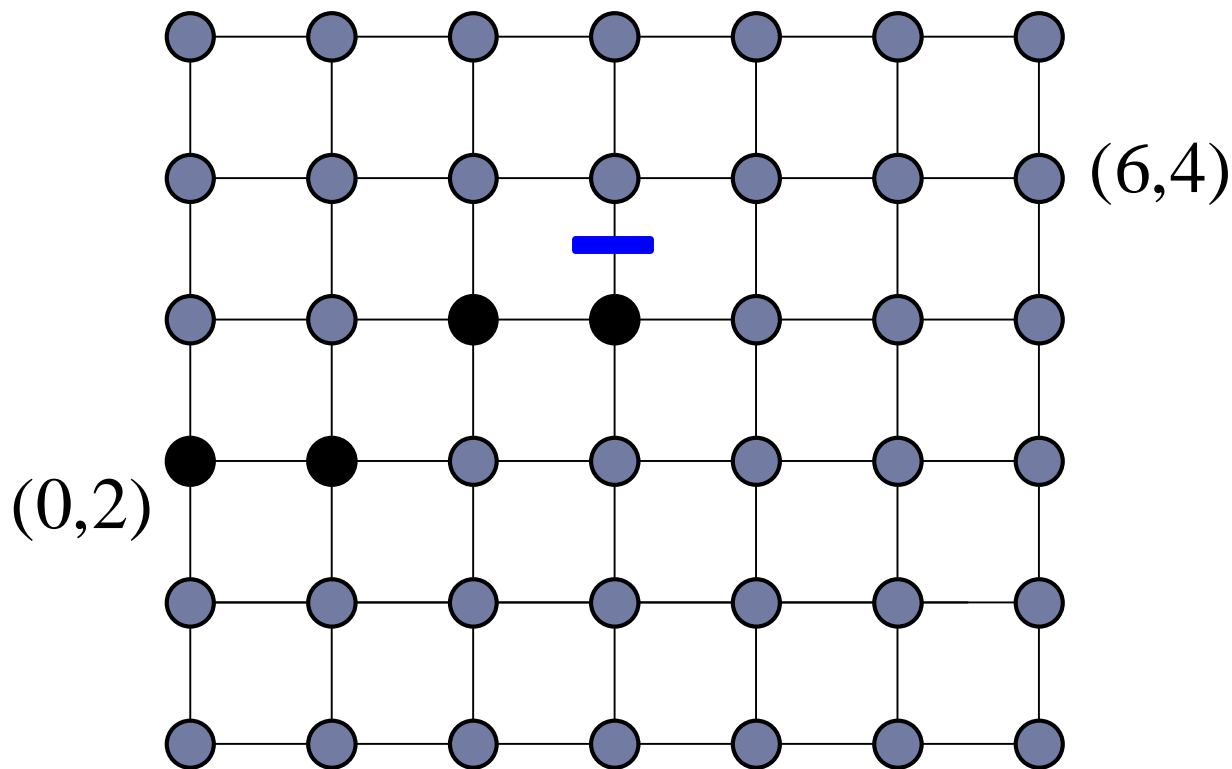
$x++$

$sum += 2c$

$draw(x, y)$



$x = 3 \quad y = 3 \quad sum = 2$



$d = 6 \quad c = -2$

Midpoint Algorithm - Παράδειγμα

$x = x_L$

$y = y_L$

$d = x_H - x_L$

$c = y_L - y_H$

$sum = 2c + d$

$draw(x, y)$

while ($x < x_H$)

if ($sum < 0$)

$sum += 2d$

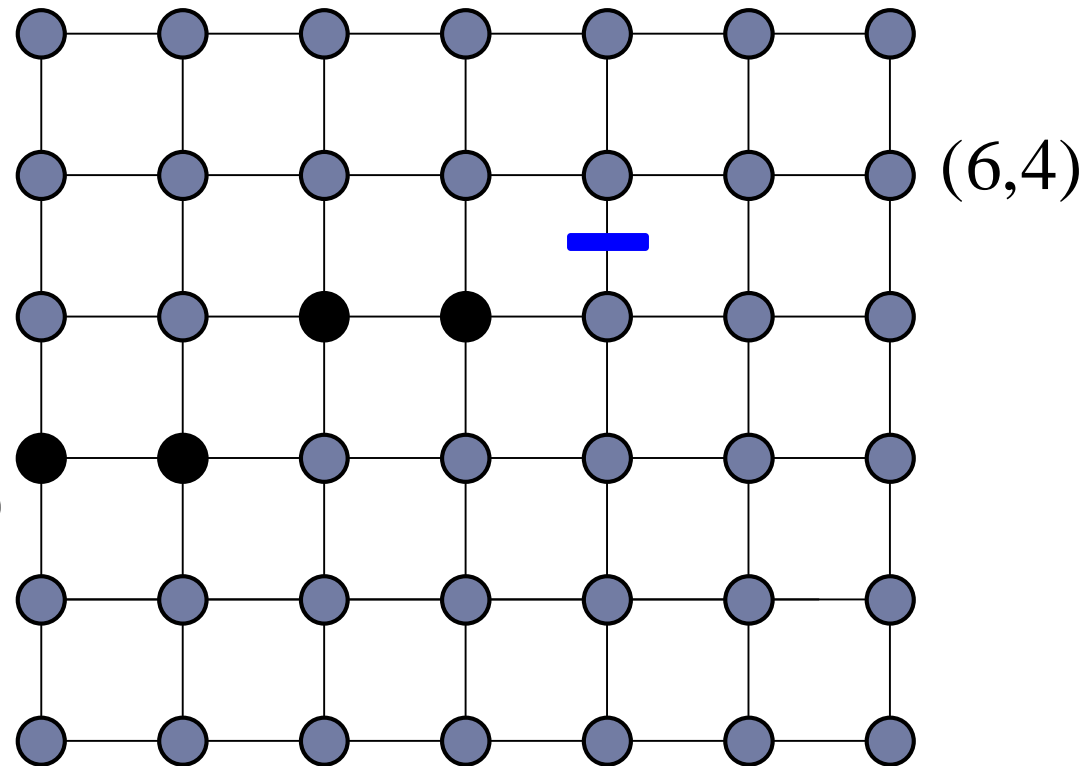
$y++$

$x++$

$sum += 2c$

$draw(x, y)$

$x=3 \quad y=3 \quad sum=2$



$d = 6 \quad c = -2$

Midpoint Algorithm - Παράδειγμα

$x = x_L$

$y = y_L$

$d = x_H - x_L$

$c = y_L - y_H$

$sum = 2c + d$

$draw(x, y)$

while ($x < x_H$)

 if ($sum < 0$)

$sum += 2d$

$y++$

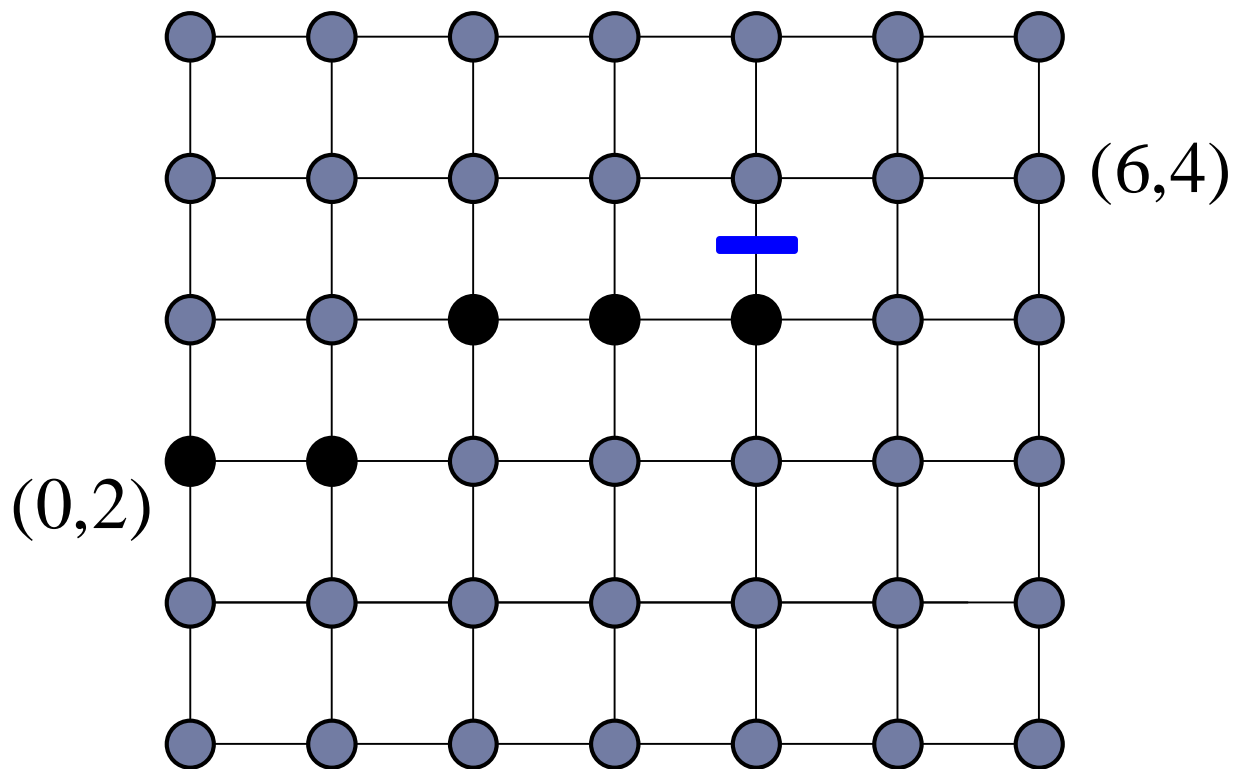
$x++$

$sum += 2c$

$draw(x, y)$



$x=4$ $y=3$ $sum=-2$



$d = 6$ $c = -2$

Midpoint Algorithm - Παράδειγμα

$x = x_L$

$y = y_L$

$d = x_H - x_L$

$c = y_L - y_H$

$sum = 2c + d$

$draw(x, y)$

$while (x < x_H)$ ←

if ($sum < 0$)

$sum += 2d$

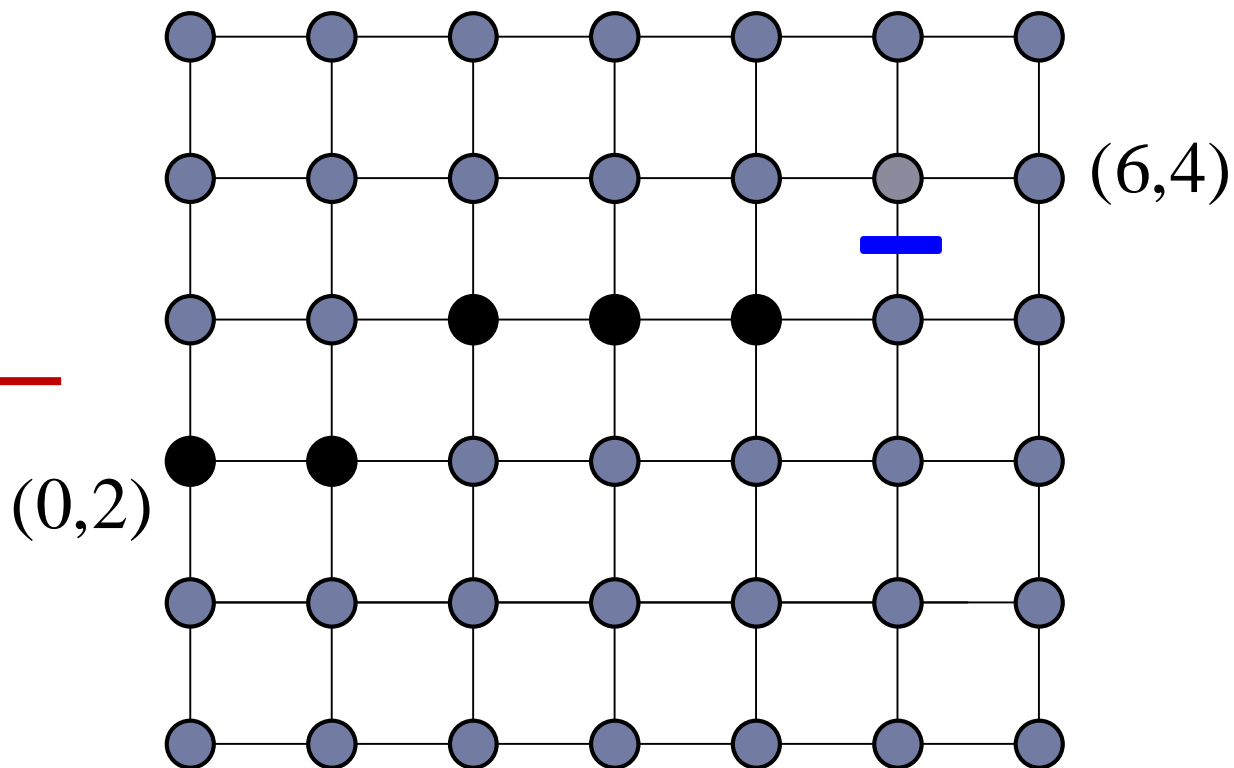
$y++$

$x++$

$sum += 2c$

$draw(x, y)$

$x = 4 \quad y = 3 \quad sum = -2$



$d = 6 \quad c = -2$

Midpoint Algorithm - Παράδειγμα

$x = x_L$

$y = y_L$

$d = x_H - x_L$

$c = y_L - y_H$

$sum = 2c + d$

$draw(x, y)$

while ($x < x_H$)

if ($sum < 0$)

$sum += 2d$

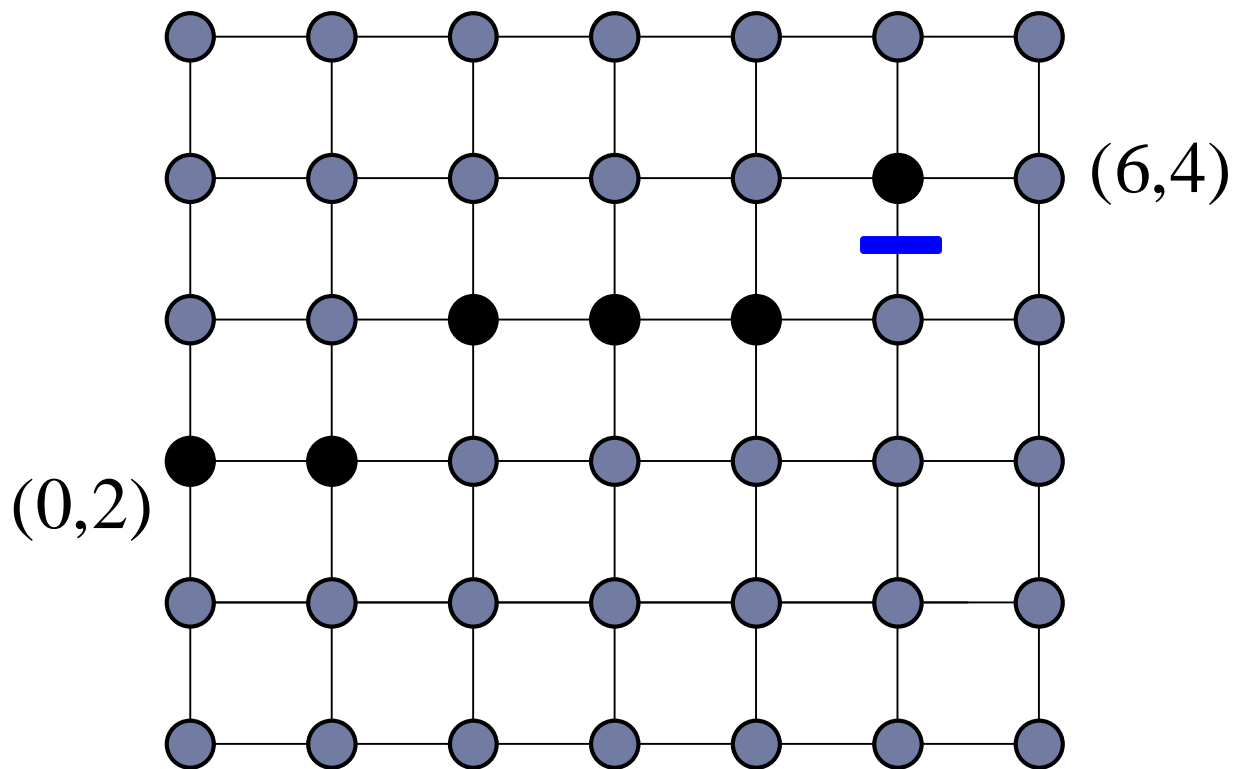
$y++$

$x++$

$sum += 2c$

$draw(x, y)$ ←

$x = 5$ $y = 4$ $sum = 6$



Midpoint Algorithm - Παράδειγμα

$x = x_L$

$y = y_L$

$d = x_H - x_L$

$c = y_L - y_H$

$sum = 2c + d$

$draw(x, y)$

while ($x < x_H$)

if ($sum < 0$)

$sum += 2d$

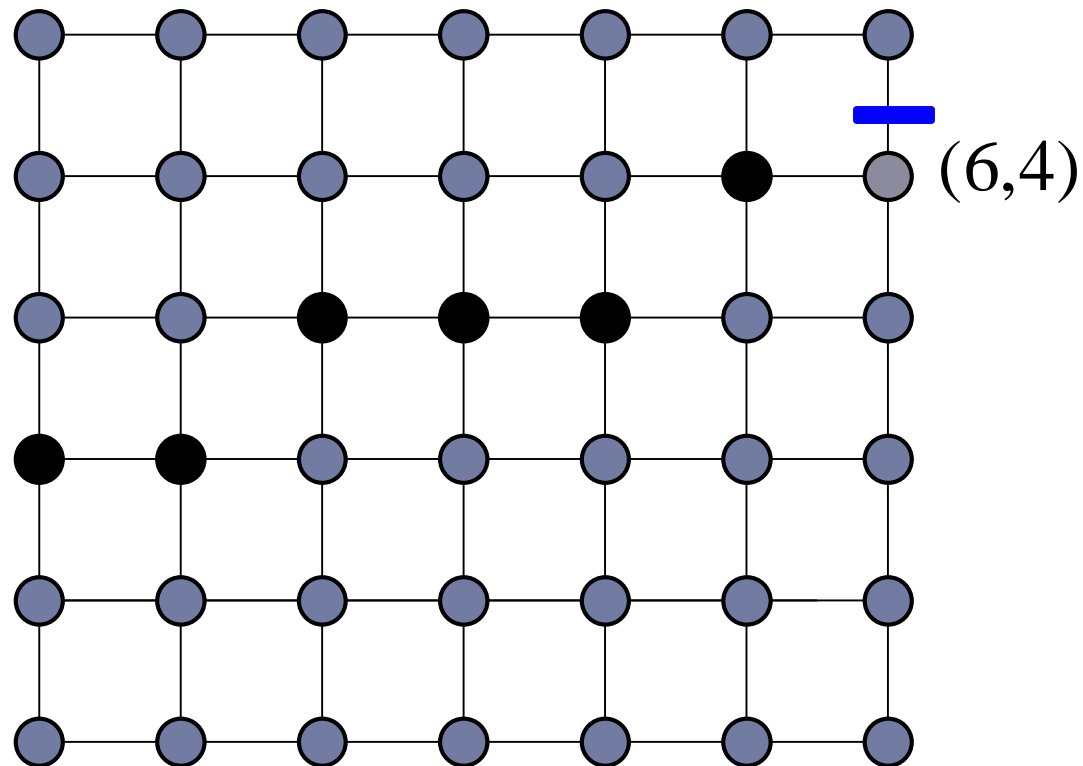
$y++$

$x++$

$sum += 2c$

$draw(x, y)$

$x = 5 \quad y = 4 \quad sum = 6$



$d = 6 \quad c = -2$

Midpoint Algorithm - Παράδειγμα

$x = x_L$

$y = y_L$

$d = x_H - x_L$

$c = y_L - y_H$

$sum = 2c + d$

$draw(x, y)$

while ($x < x_H$)

if ($sum < 0$)

$sum += 2d$

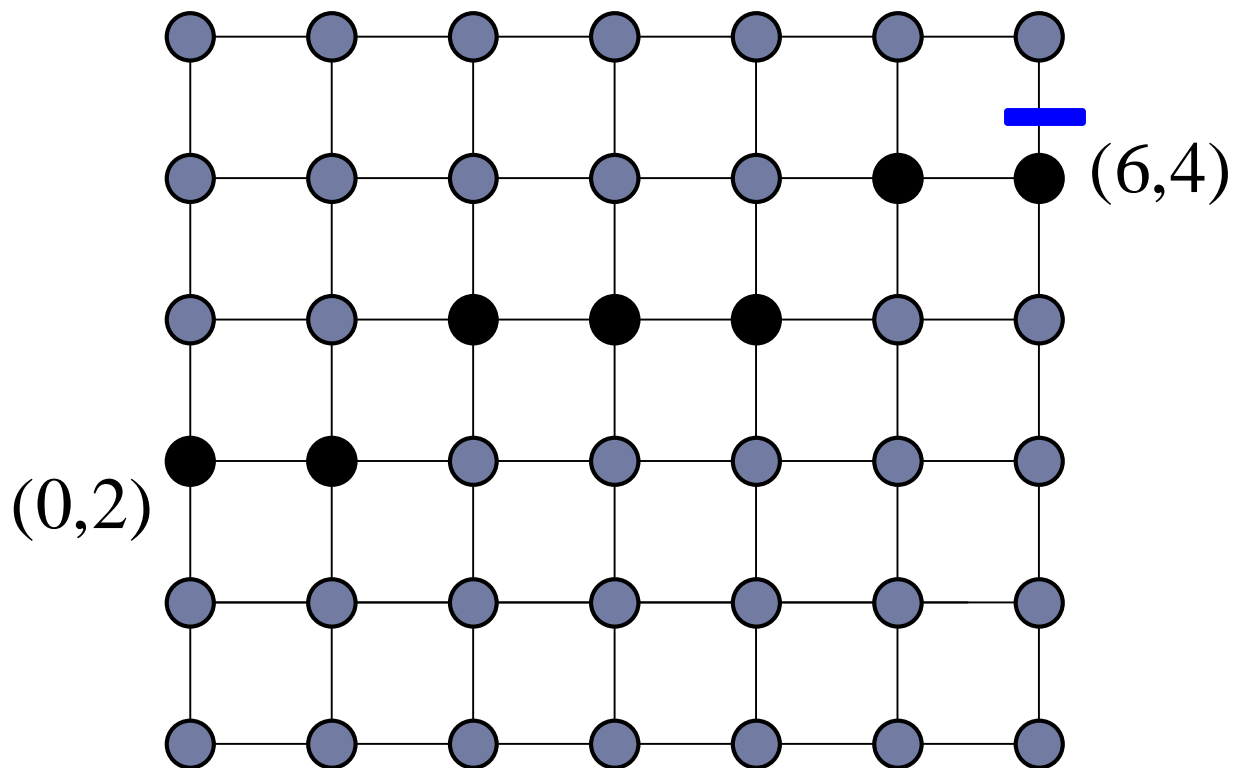
$y++$

$x++$

$sum += 2c$

$draw(x, y)$ ←

$x = 6$ $y = 4$ $sum = 2$



$d = 6$ $c = -2$

Midpoint Algorithm

- ▶ Μόνο πράξεις με ακραίους
- ▶ Ακριβώς ίδιος με τον αλγόριθμο του Bresenham (άσκηση!!!!)
- ▶ Επεκτείνεται και σε άλλα σχήματα (κύκλος, έλλειψη κλπ.)

Bresenham Algorithm για ευθείες ($0 < m < 1$)

```
x0=xL
y0=yL
dx=xH - xL
dy= yH - yL
p0=2dy-dx
k=0
draw(xk,yk)
while ( xk < xH)
    if ( pk < 0 )
        pk+1 = pk+2dy
        draw(xk+1 ,yk)
    else
        pk+1 = pk+2dy-2dx
        draw(xk+1 ,yk+1)

    k++
end
```