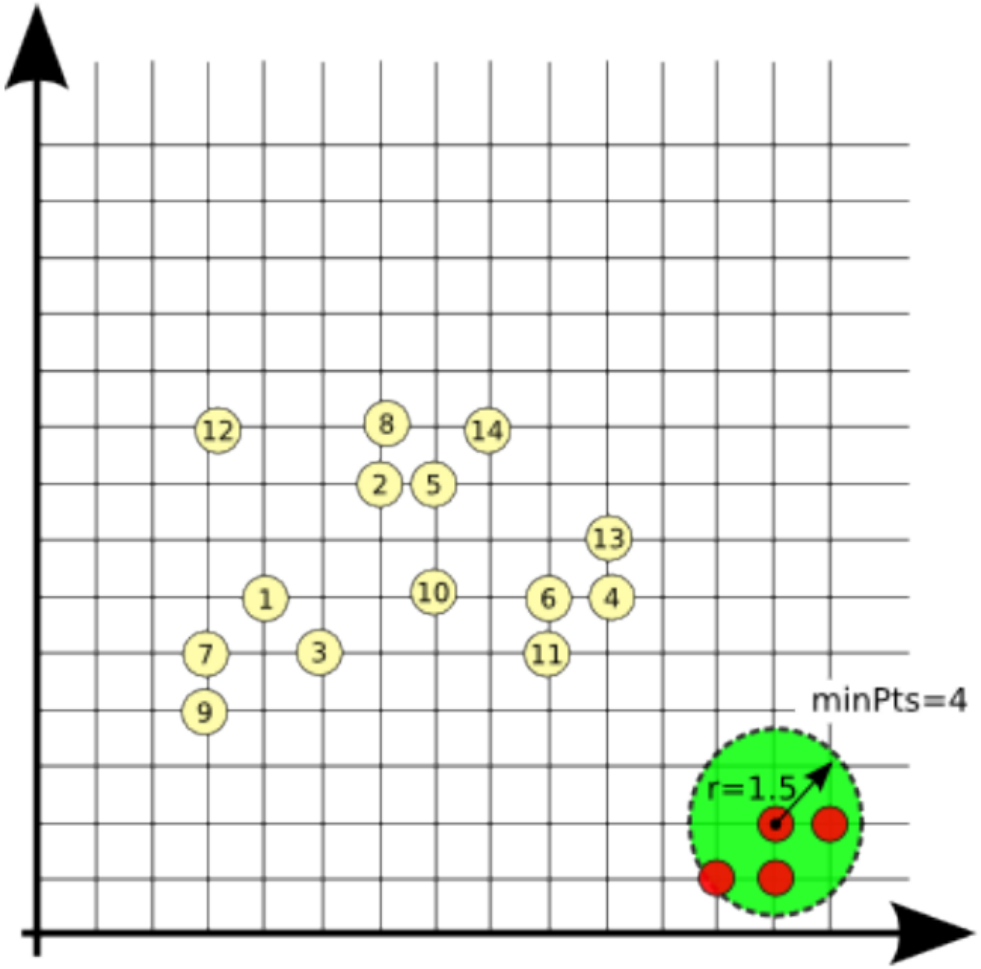
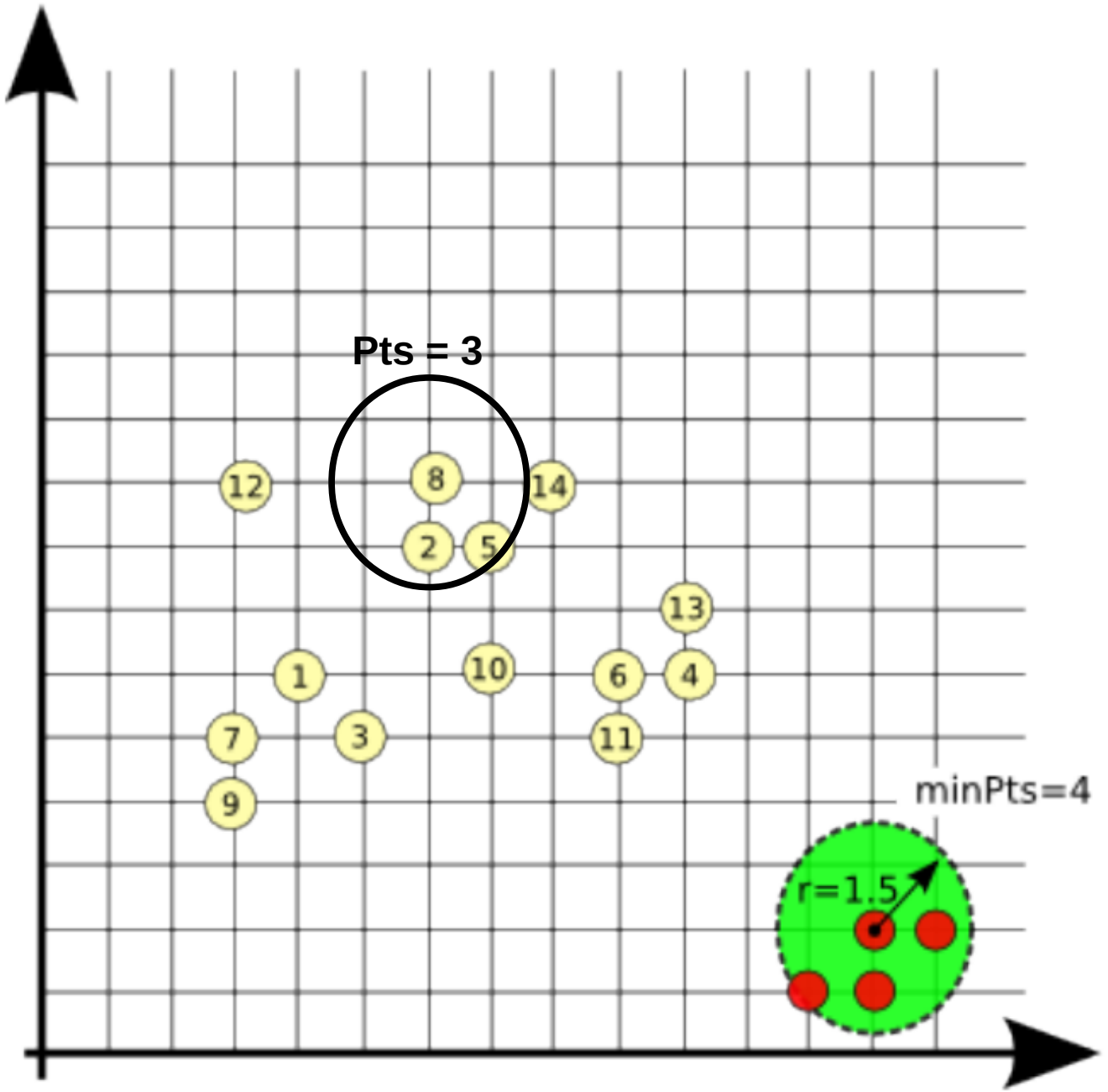
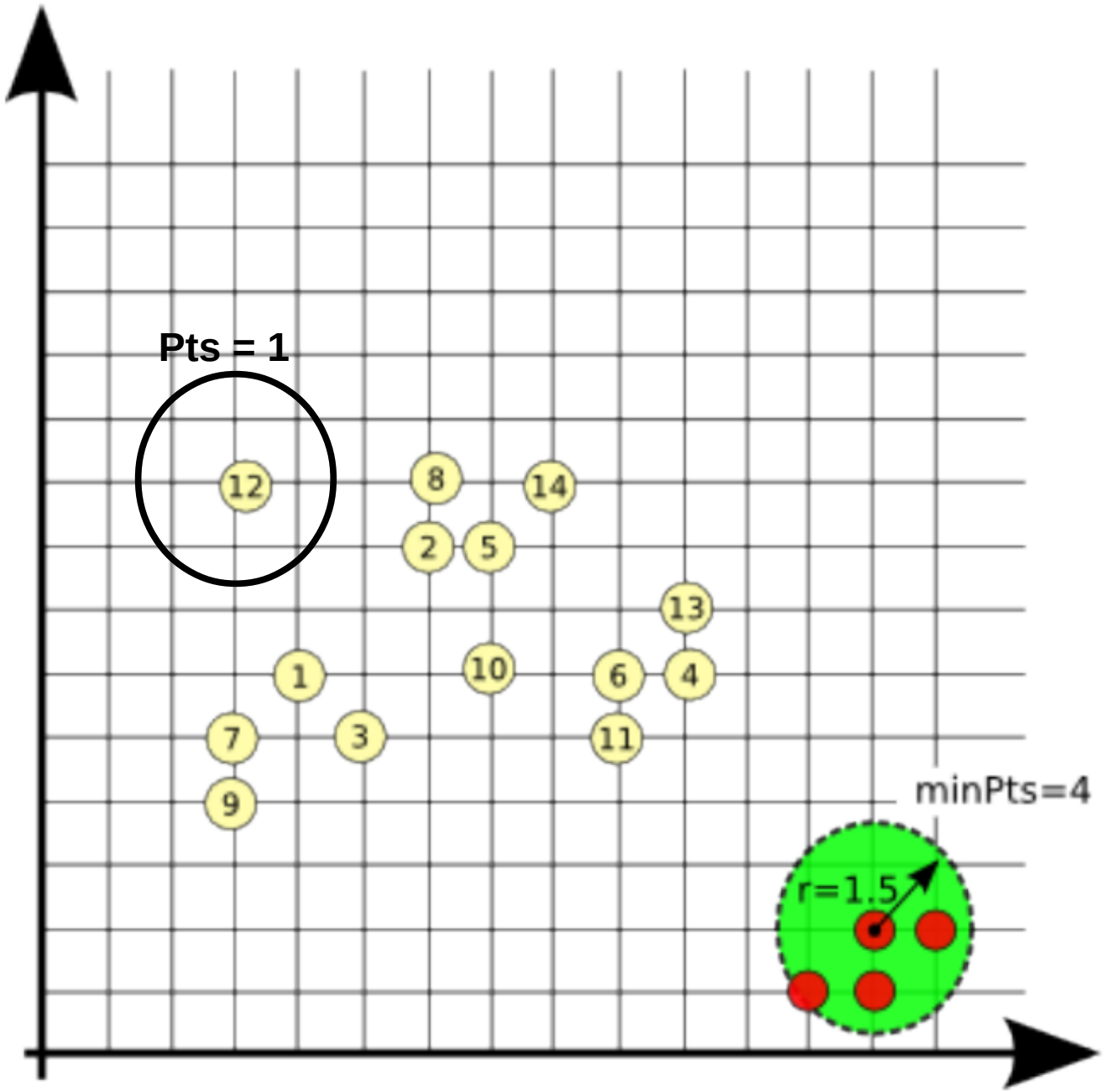
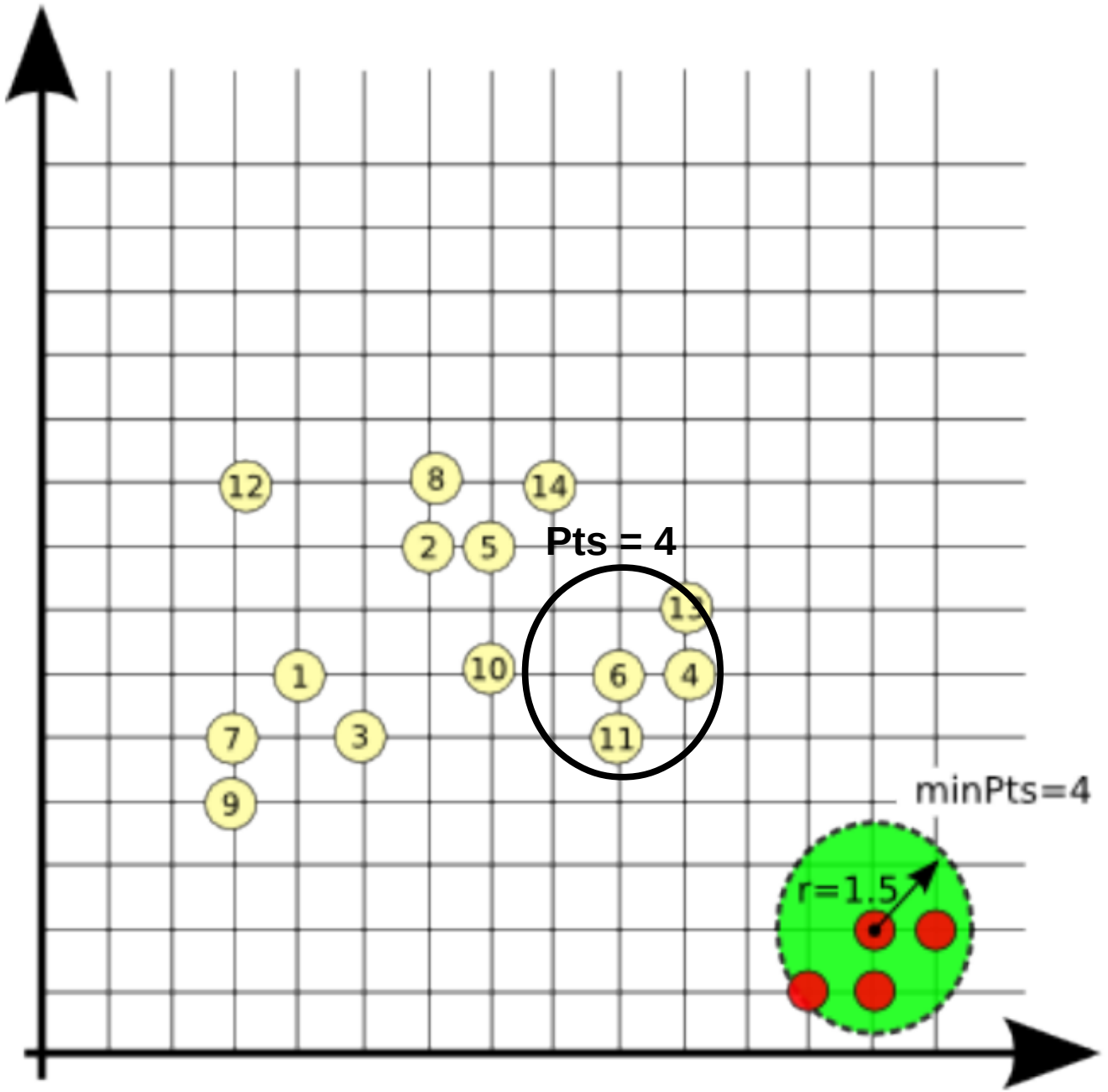
Distance-based outlier detection

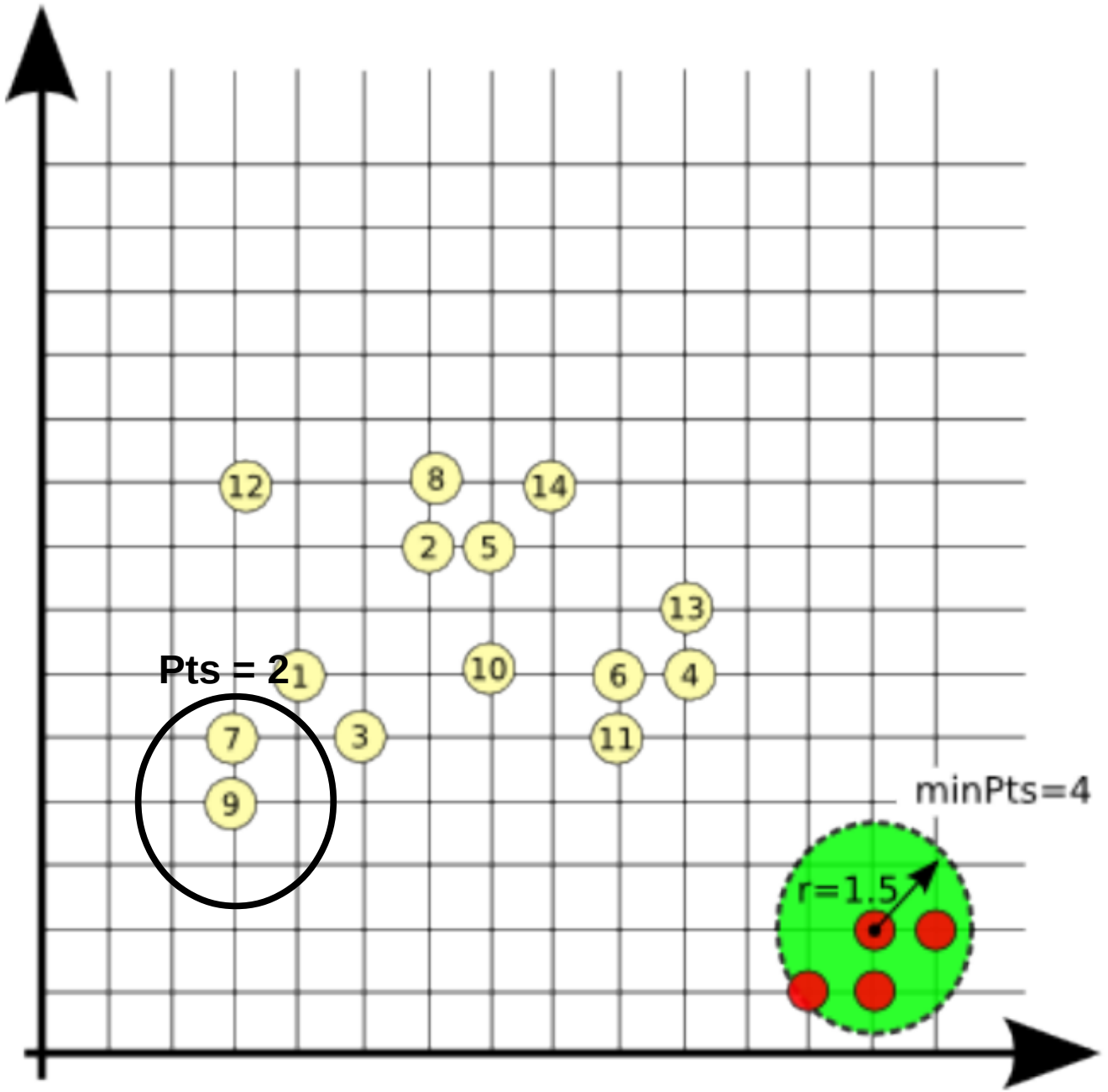
- Given the dataset of the right, find the outliers according to the basic DB(ε,π)-Outliers method with:
  - ε = 1.5
  - π = 0.28 = 4 points (including the reference point we are analyzing)

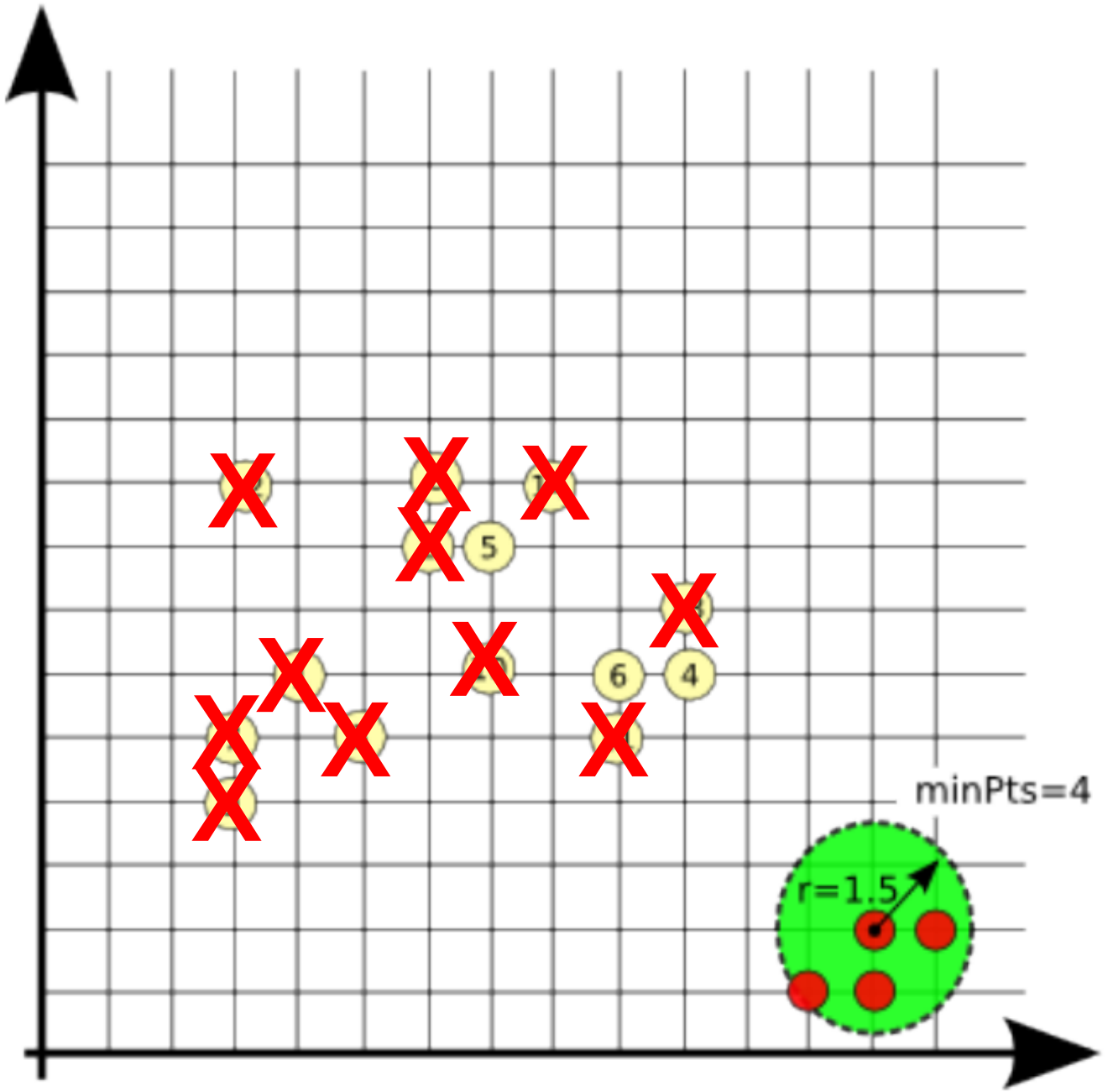



Pts = 3
12
8
14
2
5
13
1
10
6
4
7
3
11
9
minPts=4
r=1.5


Pts = 1
12
8
14
2
5
13
1
10
6
4
7
3
11
9
minPts=4
r=1.5

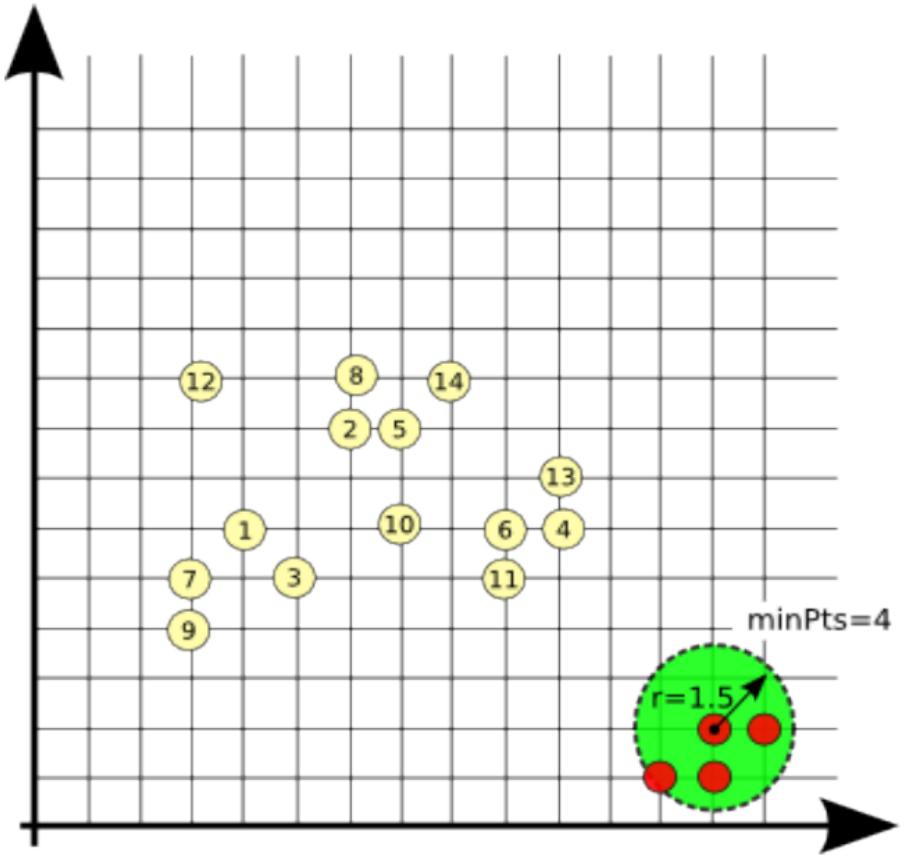

12
8
14
2
5
Pts = 4
1
10
6
4
7
3
11
9
minPts=4
r=1.5


12
8
14
2
5
13
Pts = 2
1
10
6
4
7
3
11
9
minPts=4
r=1.5


5
6
4
minPts=4
r=1.5

Density-based outlier detection

- Compute LOF score for points 5, 1, 10 of the dataset:
- K = 3
  - The reference point is not considered
- Simplify: use distance in place of reachability distance (reach_dist(p,o))



$$lrd_k(p) = 1 / \left(\frac{\sum_{o \in kNN(p)} reach\text{-}dist_k(p,o)}{Card(kNN(p))} \right)$$

$$LOF_k(p) = \frac{\sum_{o \in kNN(p)} \frac{lrd_k(o)}{lrd_k(p)}}{Card(kNN(p))}$$

$NN_3(5)$ = (2), (8), (14)

Lrd(5) = 1/((1 + 1.41 + 1.41)/3) = 0.78

→ Lrd(2) = 1/((1 + 1 + 2.23 + 2.23)/4) = 0.62

→ Lrd(8) = 1/((1 + 1.41 + 2)/3) = 0.68

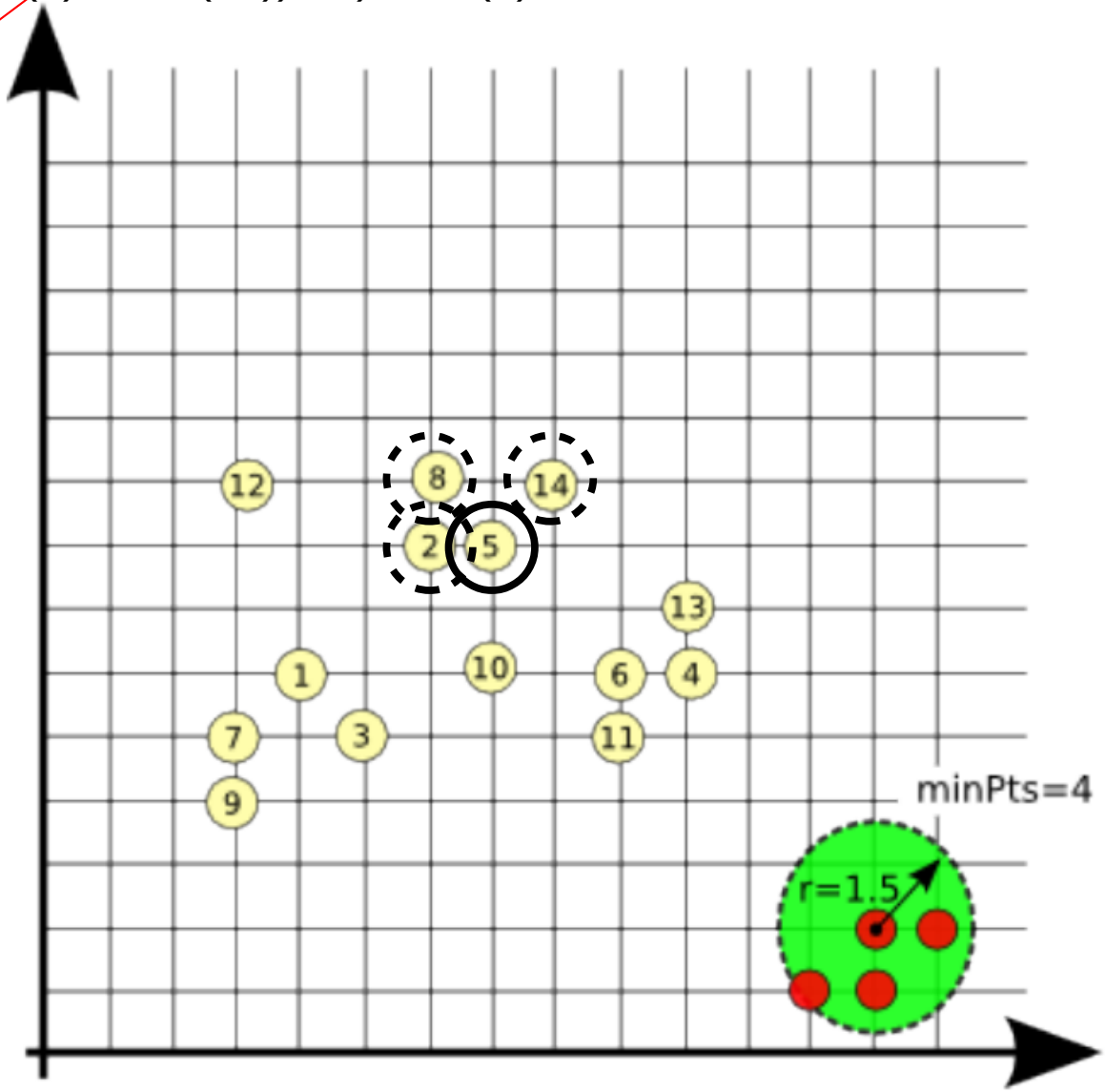
→ Lrd(14) = 1/((2 + 1.41 + 2.23)/3) = 0.53

LOF(5) = ((Lrd(2) + Lrd(8) + Lrd(14)) /3) / Lrd(5) = 0.61 / 0.78 = **0.78**

$$lrd_k(p) = 1/\left(\frac{\sum_{o \in kNN(p)} reach\text{-}dist_k(p,o)}{Card(kNN(p))}\right)$$

$$LOF_k(p) = \frac{\sum_{o \in kNN(p)} \frac{lrd_k(o)}{lrd_k(p)}}{Card(kNN(p))}$$

Notice that (2) has 4 points in its 3-NN set, because (14) and (10) have the same distance.
In case of ties we take all the points having that same distance.



$NN_3(1)$ = (3), (7), (9)

Lrd(1) = 1/((1.41 + 1.41 + 2.23)/3) = 0.59

→ Lrd(3) = 1/((1.41 + 2 + 2.23 + 2.23)/4) = 0.50

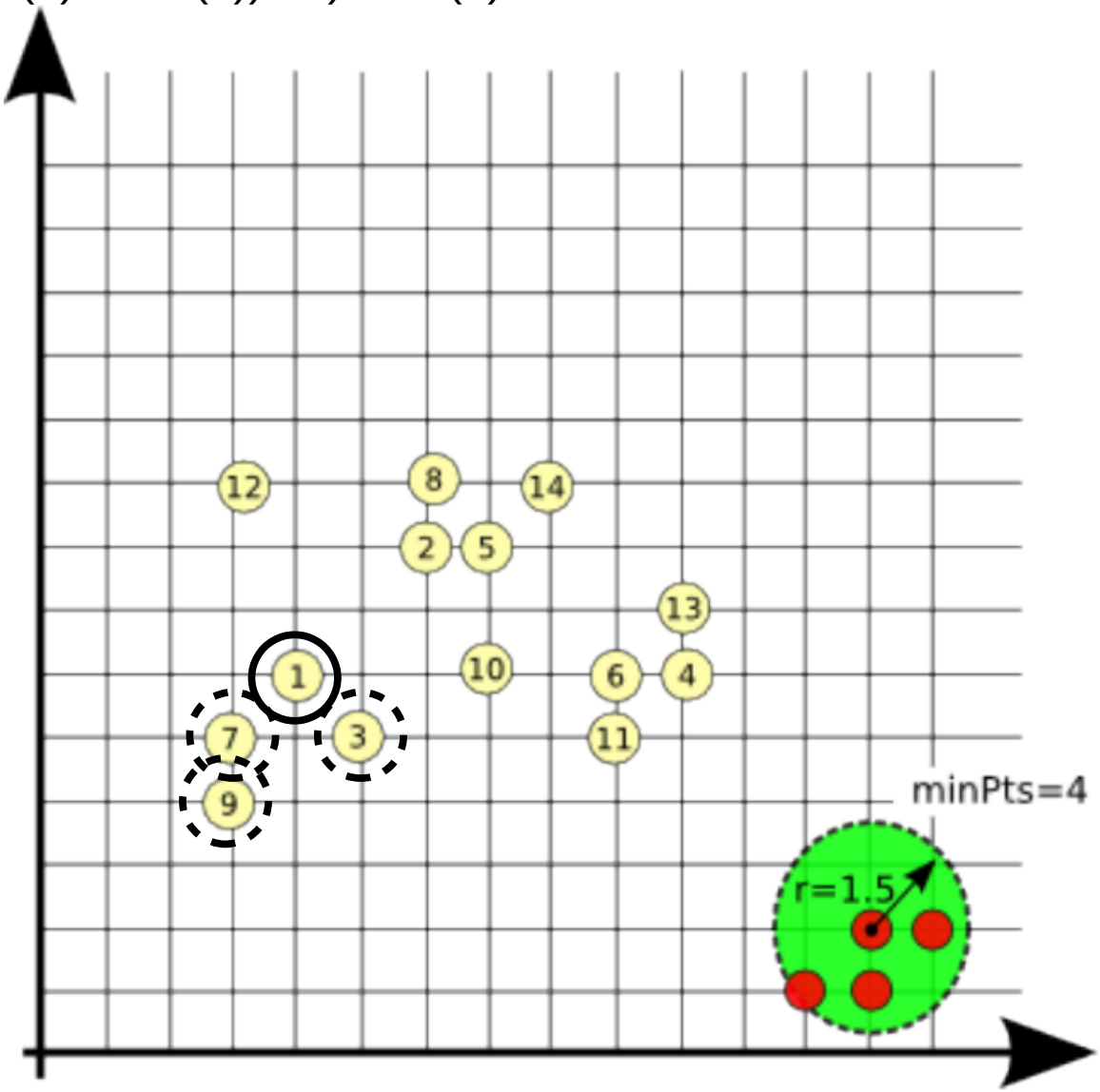
→ Lrd(7) = 1/((1 + 1.41 + 2)/3) = 0.68

→ Lrd(9) = 1/((1 + 2.23 + 2.23)/3) = 0.55

$$lrd_k(p) = 1/\left(\frac{\sum_{o \in kNN(p)} reach\text{-}dist_k(p,o)}{Card(kNN(p))}\right)$$

$$LOF_k(p) = \frac{\sum_{o \in kNN(p)} \frac{lrd_k(o)}{lrd_k(p)}}{Card(kNN(p))}$$

LOF(5) = ((Lrd(3) + Lrd(7) + Lrd(9)) /3) / Lrd(1) = 0.57 / 0.59 = **0.96**



$NN_3(10) = (2), (5), (6), (11), (3)$

Lrd(10) = 1/((2 + 2.23 + 2 + 2.23 + 2.23)/5) = 0.46

→ Lrd(2) = 1/((1 + 1 + 2.23 + 2.23)/4) = 0.62

→ Lrd(5) = 1/((1 + 1.41 + 1.41)/3) = 0.78

→ Lrd(6) = 1/((1 + 1 + 1.41)/3) = 0.87

→ Lrd(11) = 1/((1 + 1.41 + 2.23 + 2.23)/4) = 0.58

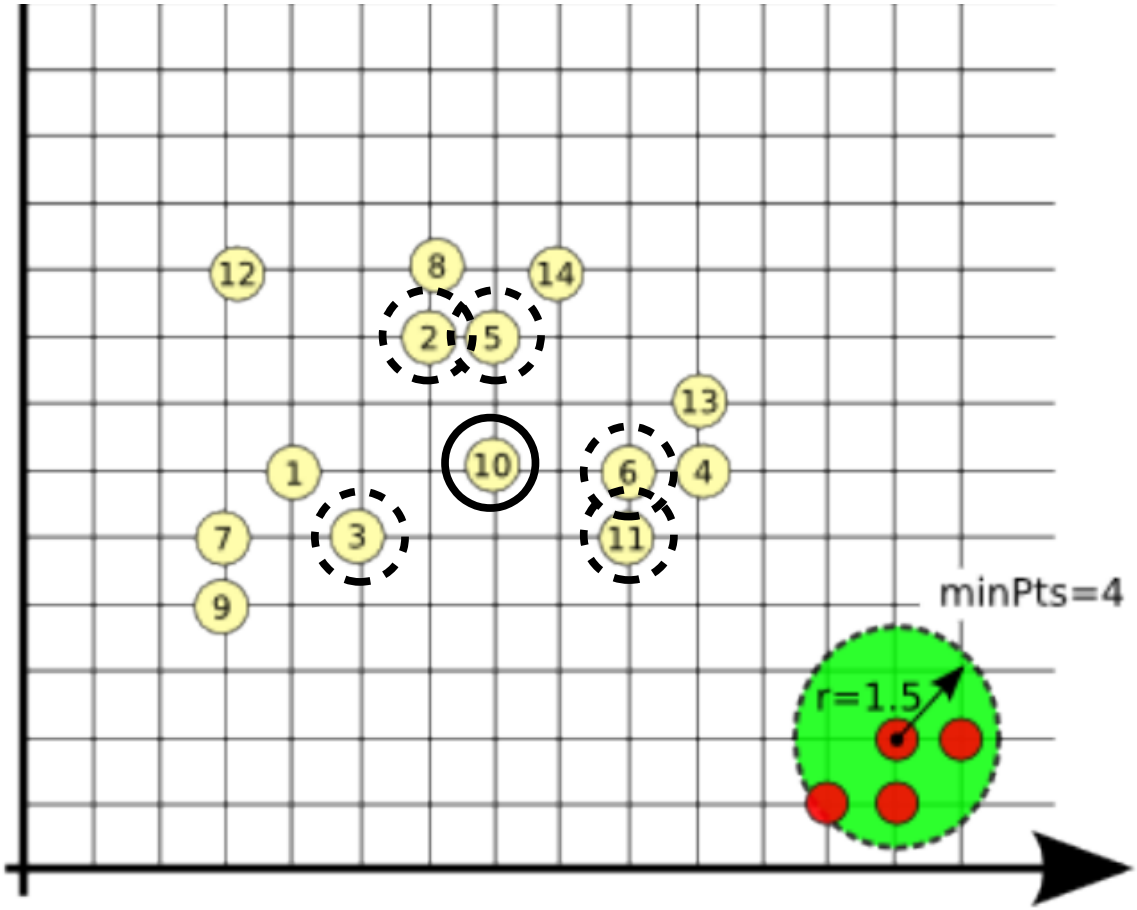
→ Lrd(3) = 1/((1.41 + 2 + 2.23 + 2.23)/4) = 0.50

$$lrd_k(p) = 1/\left(\frac{\sum_{o \in kNN(p)} reach\text{-}dist_k(p,o)}{Card(kNN(p))}\right)$$

$$LOF_k(p) = \frac{\sum_{o \in kNN(p)} \frac{lrd_k(o)}{lrd_k(p)}}{Card(kNN(p))}$$

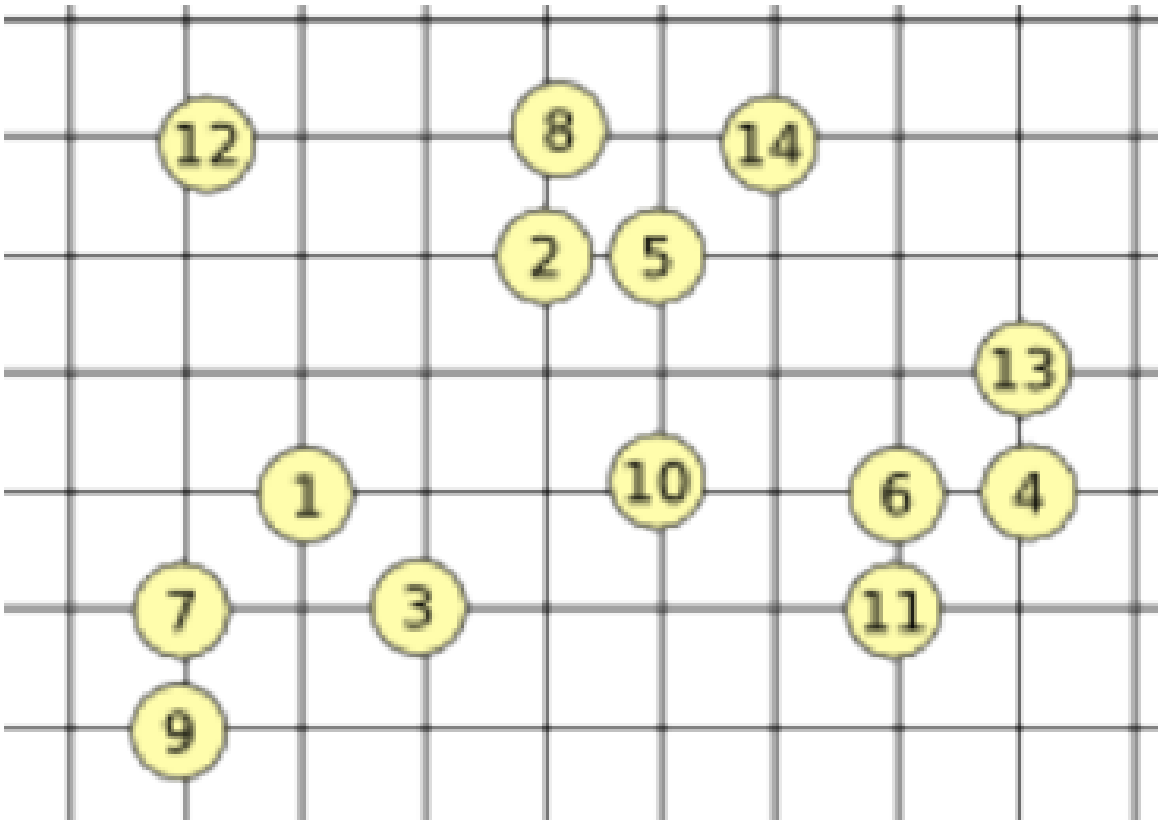
LOF(10) = ((Lrd(2) + Lrd(5) + Lrd(6) + Lrd(11) + Ldr(3)) /5) / Lrd(10) = 0.67 / 0.46 = **1.45**

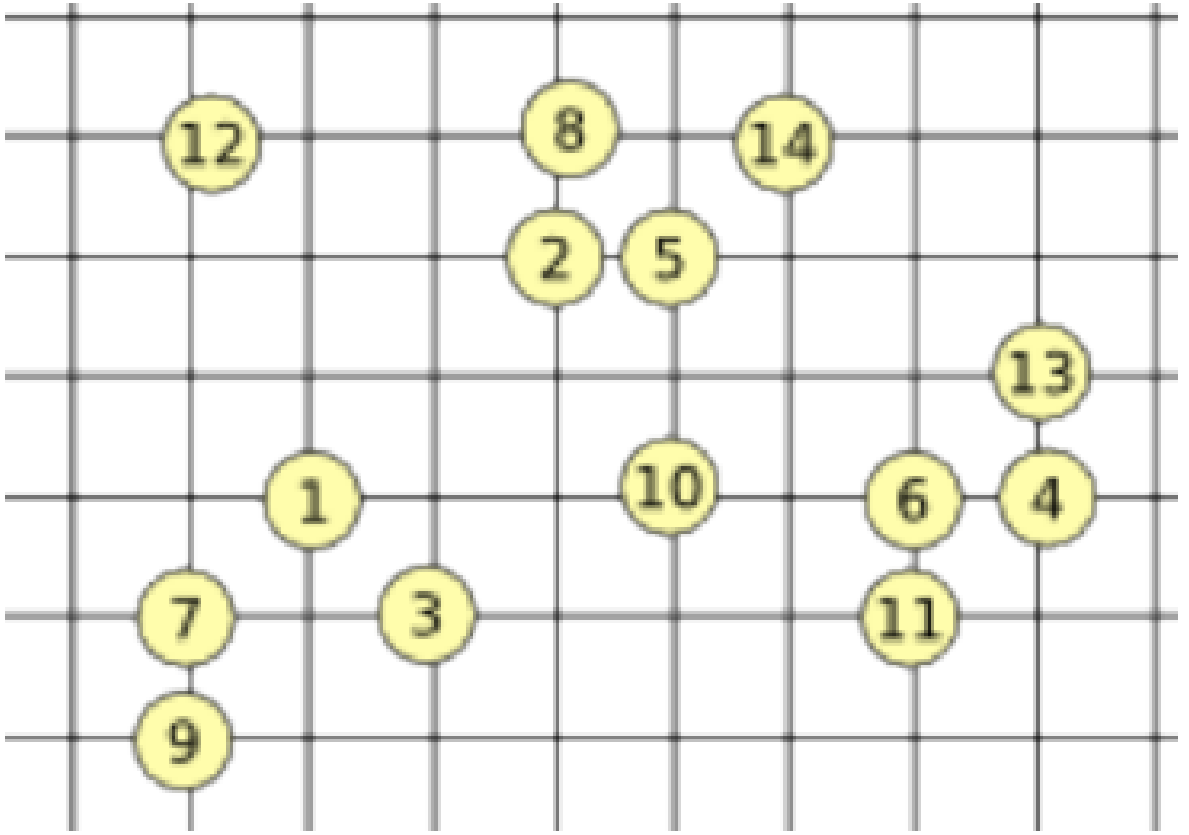
Here (10) has 3 points in a tie for the 3rd place in the 3-NN set (2, 3, 11), resulting in 5 NNs.

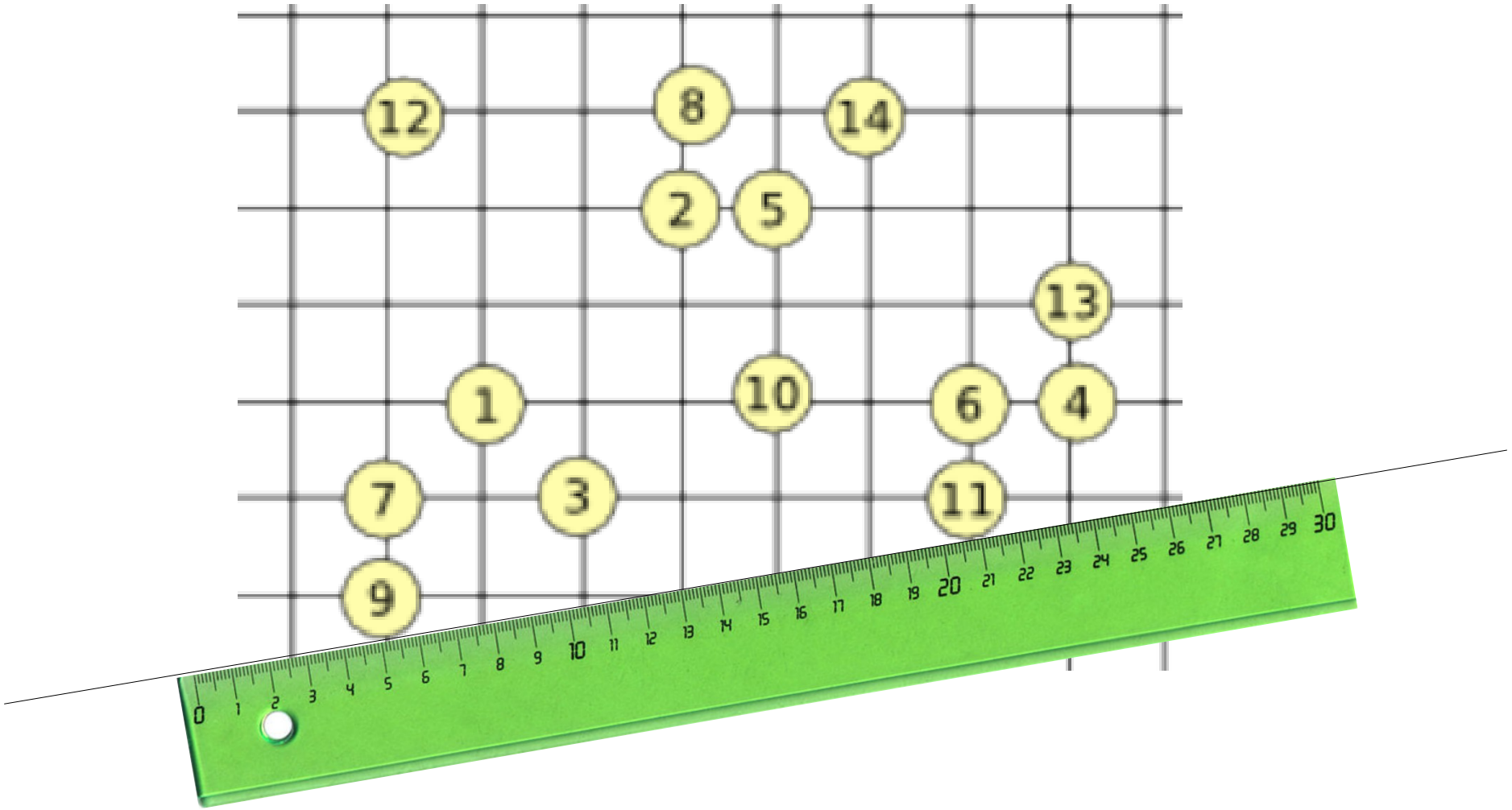


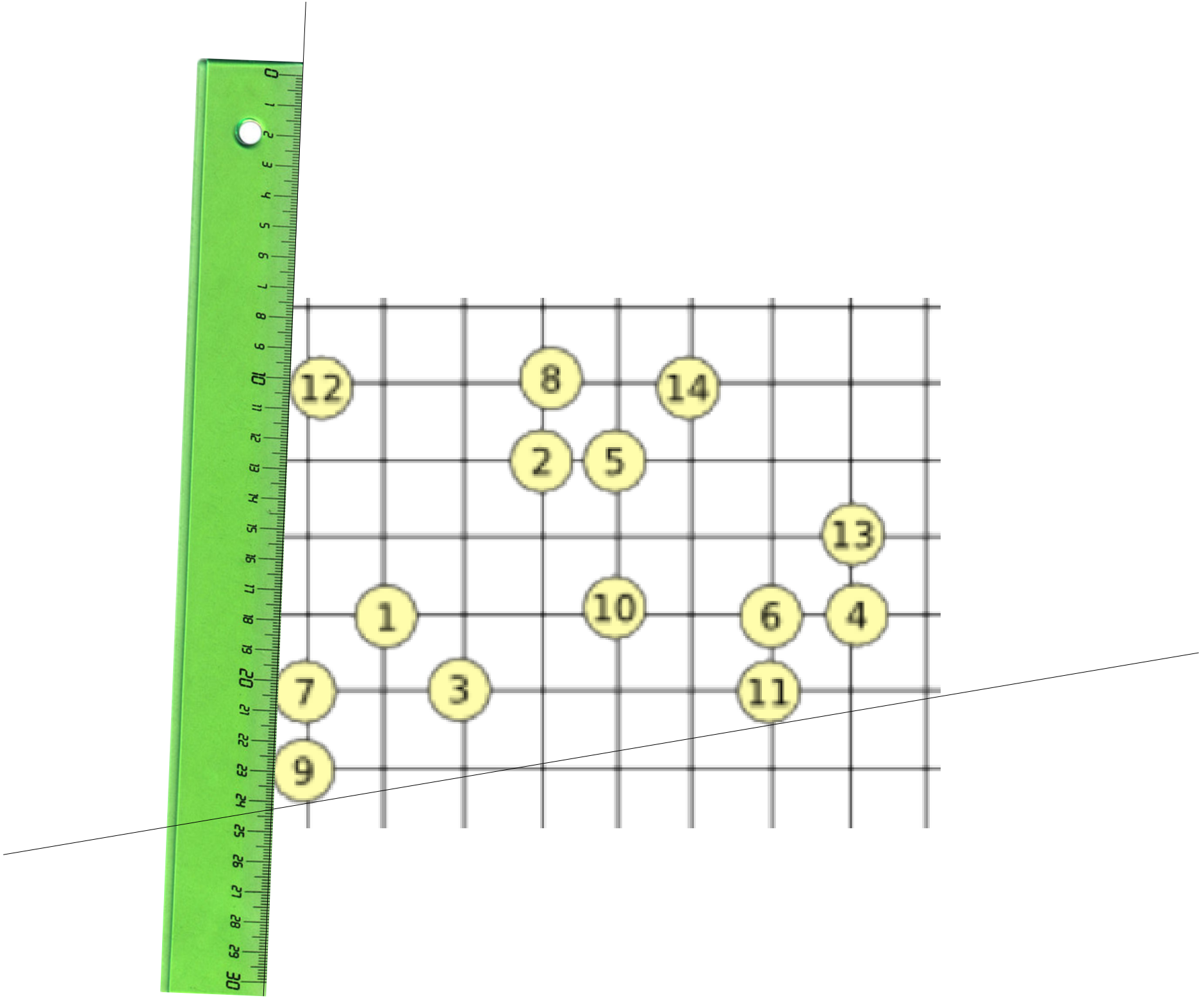
Depth-based outliers

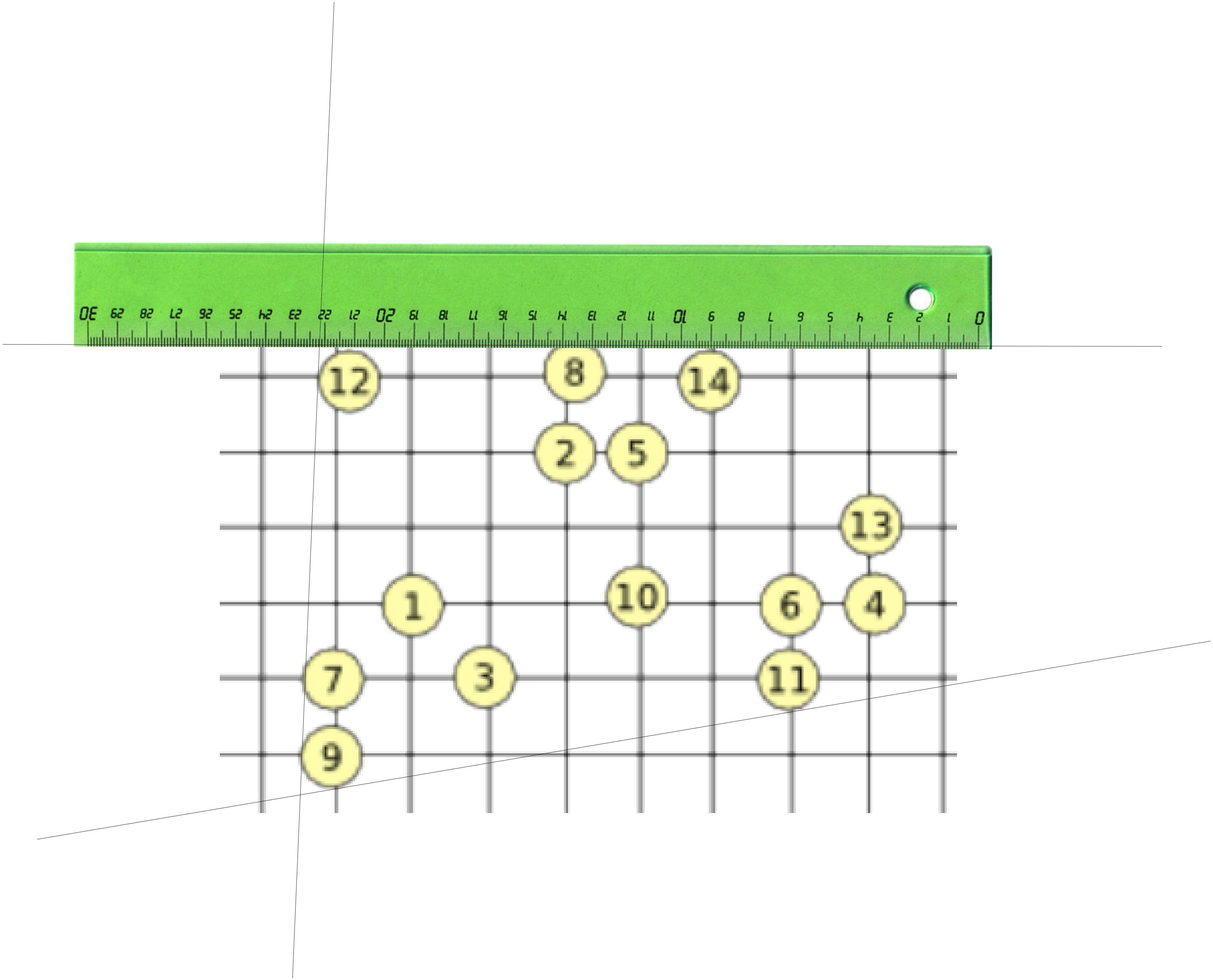
- Compute the depth score of each point

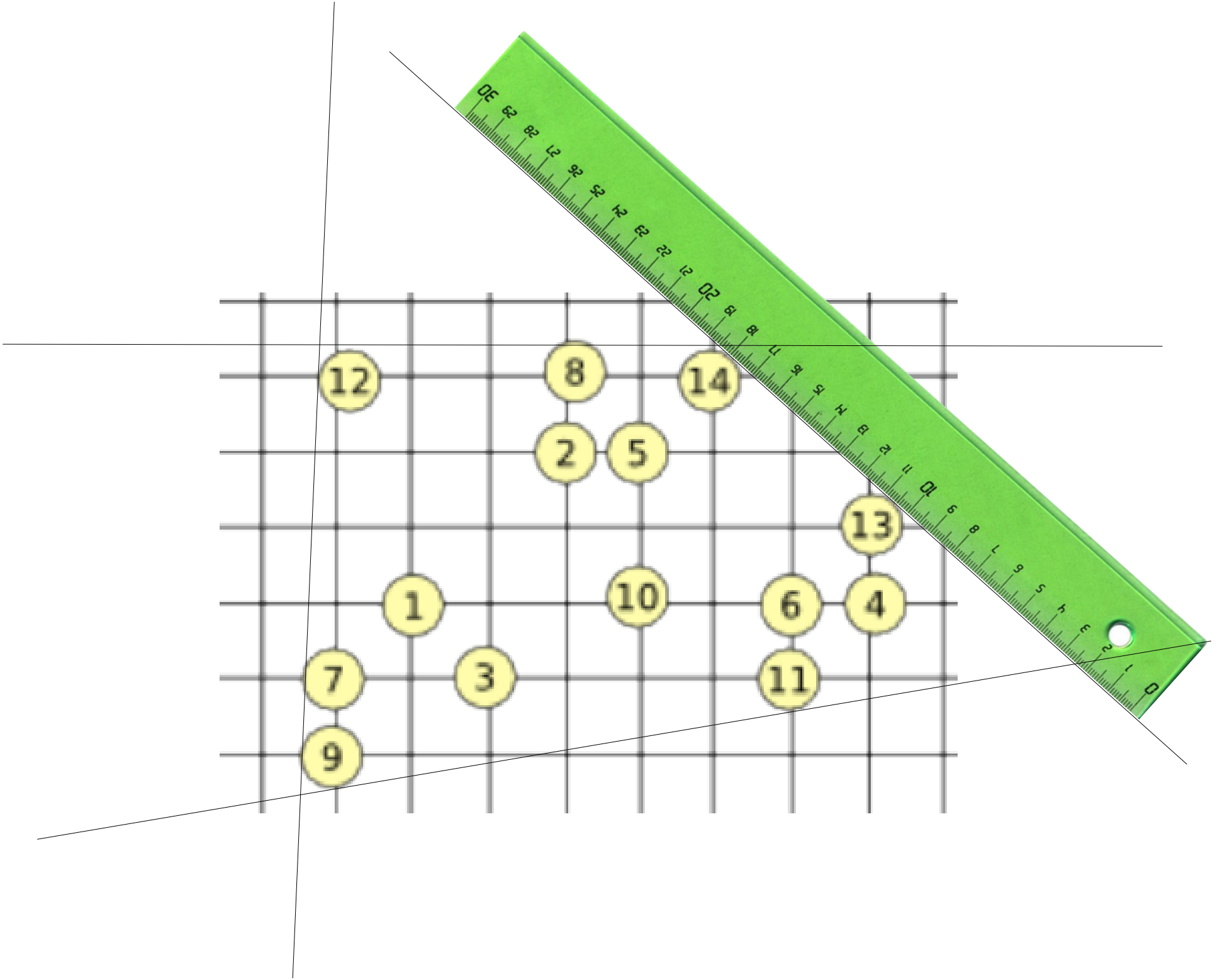


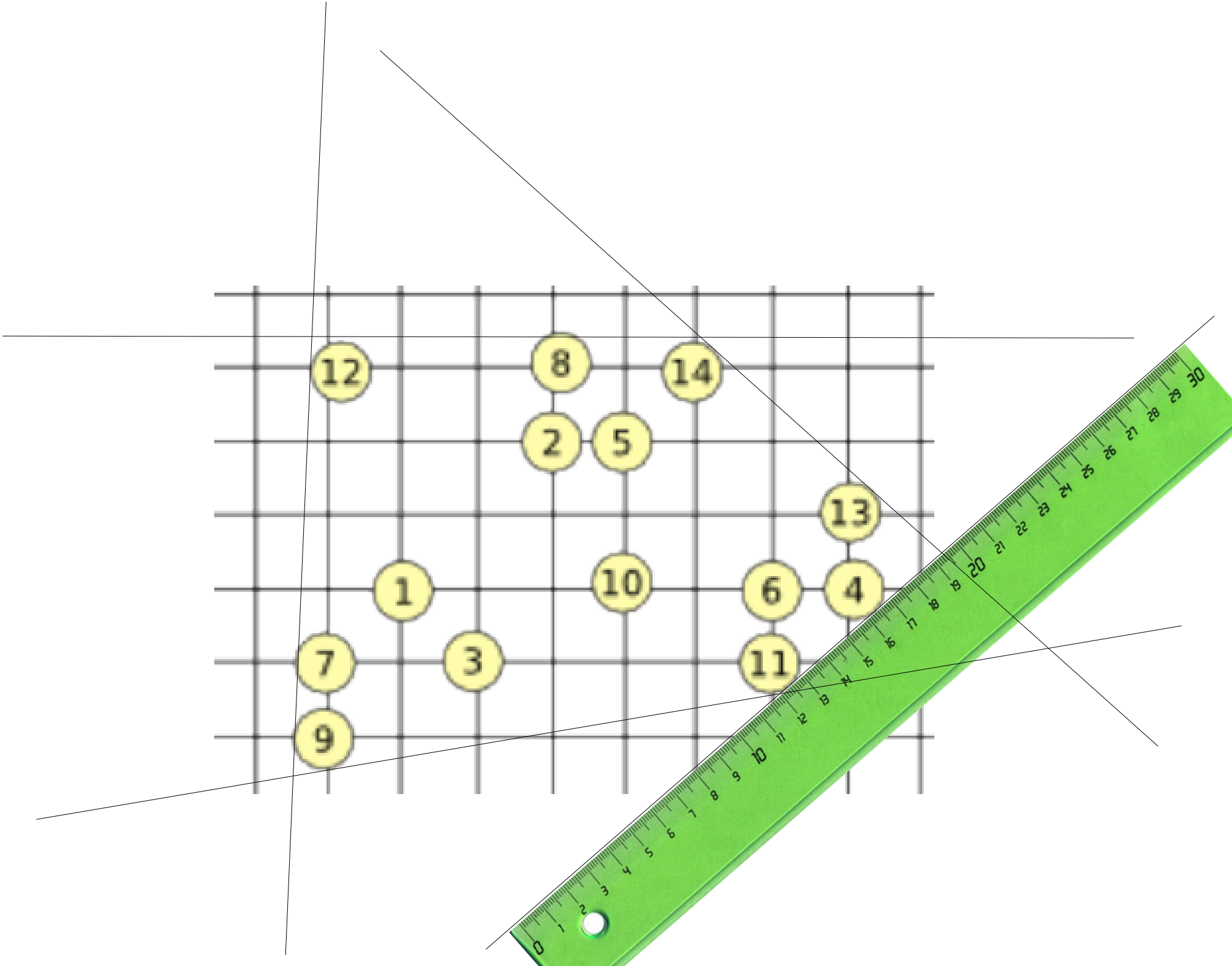

12
8
14
2
5
13
1
10
6
4
7
3
11
9

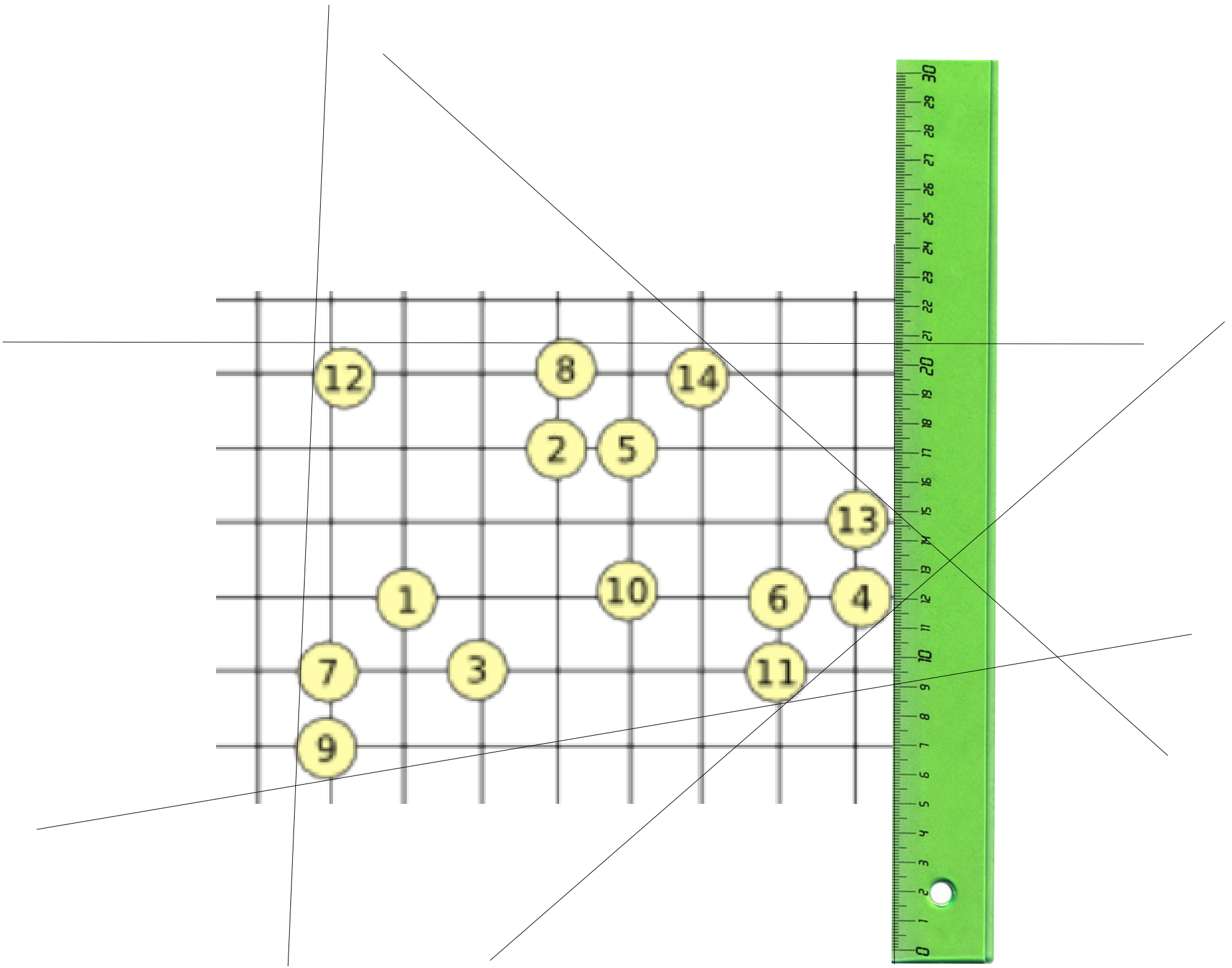

12
8
14
2
5
13
1
10
6
4
7
3
11
9
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30

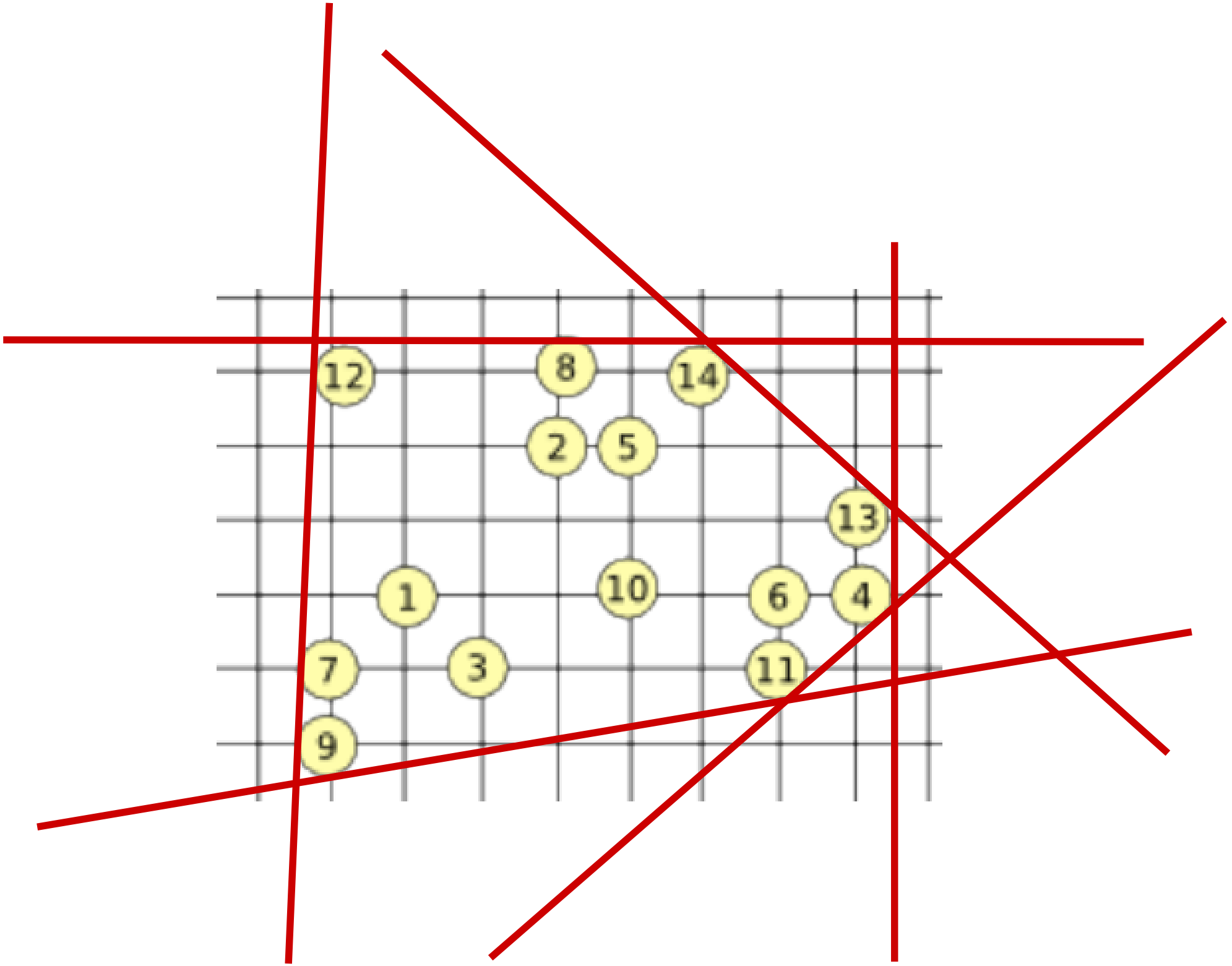

12
8
14
2
5
13
1
10
6
4
7
3
11
9


12
8
14
2
5
13
1
10
6
4
7
3
11
9

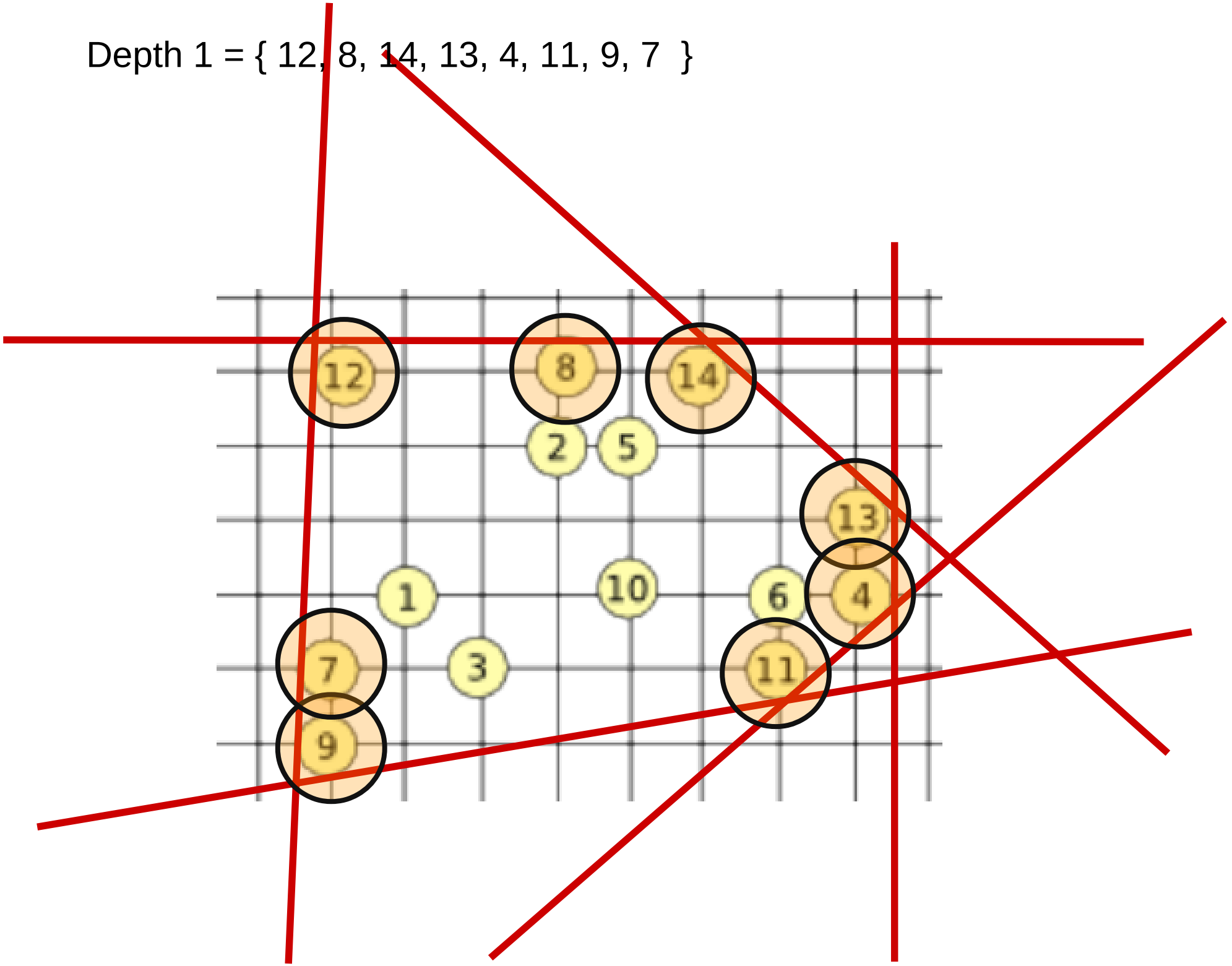

12
8
14
2
5
13
1
10
6
4
7
3
11
9


12
8
14
2
5
13
1
10
6
4
7
3
11
9

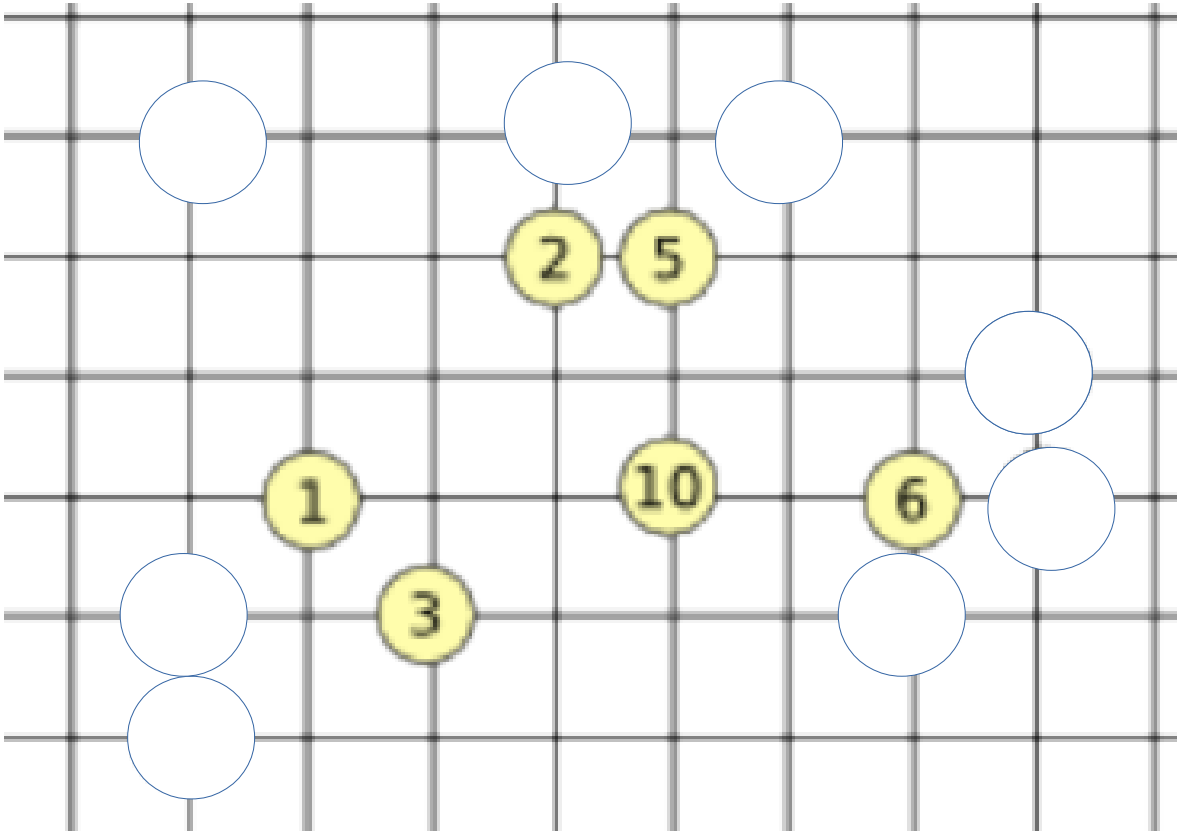

12
8
14
2
5
13
1
10
6
4
7
3
11
9


12
8
14
2
5
13
1
10
6
4
7
3
11
9

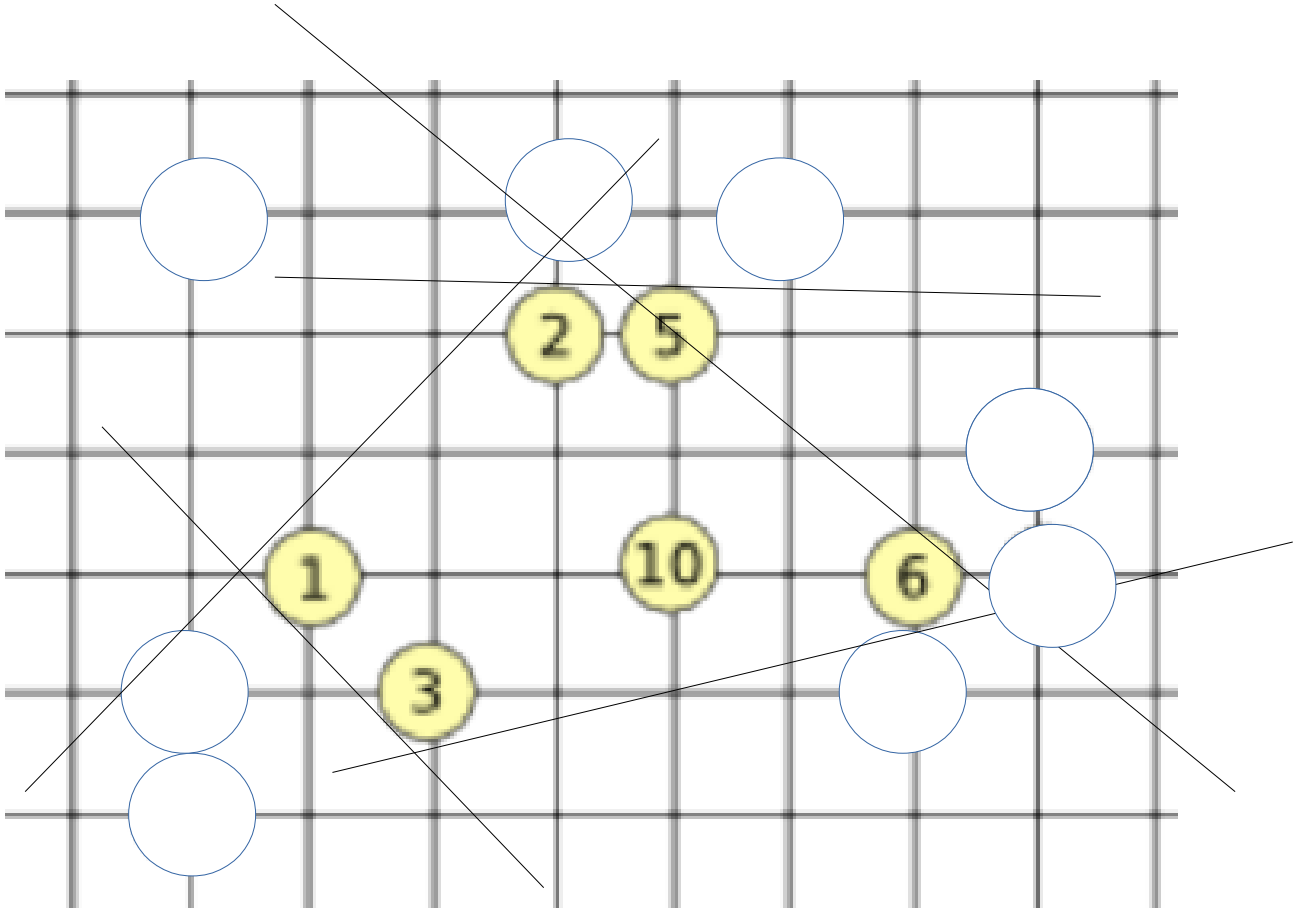
Depth 1 = { 12, 8, 14, 13, 4, 11, 9, 7 }



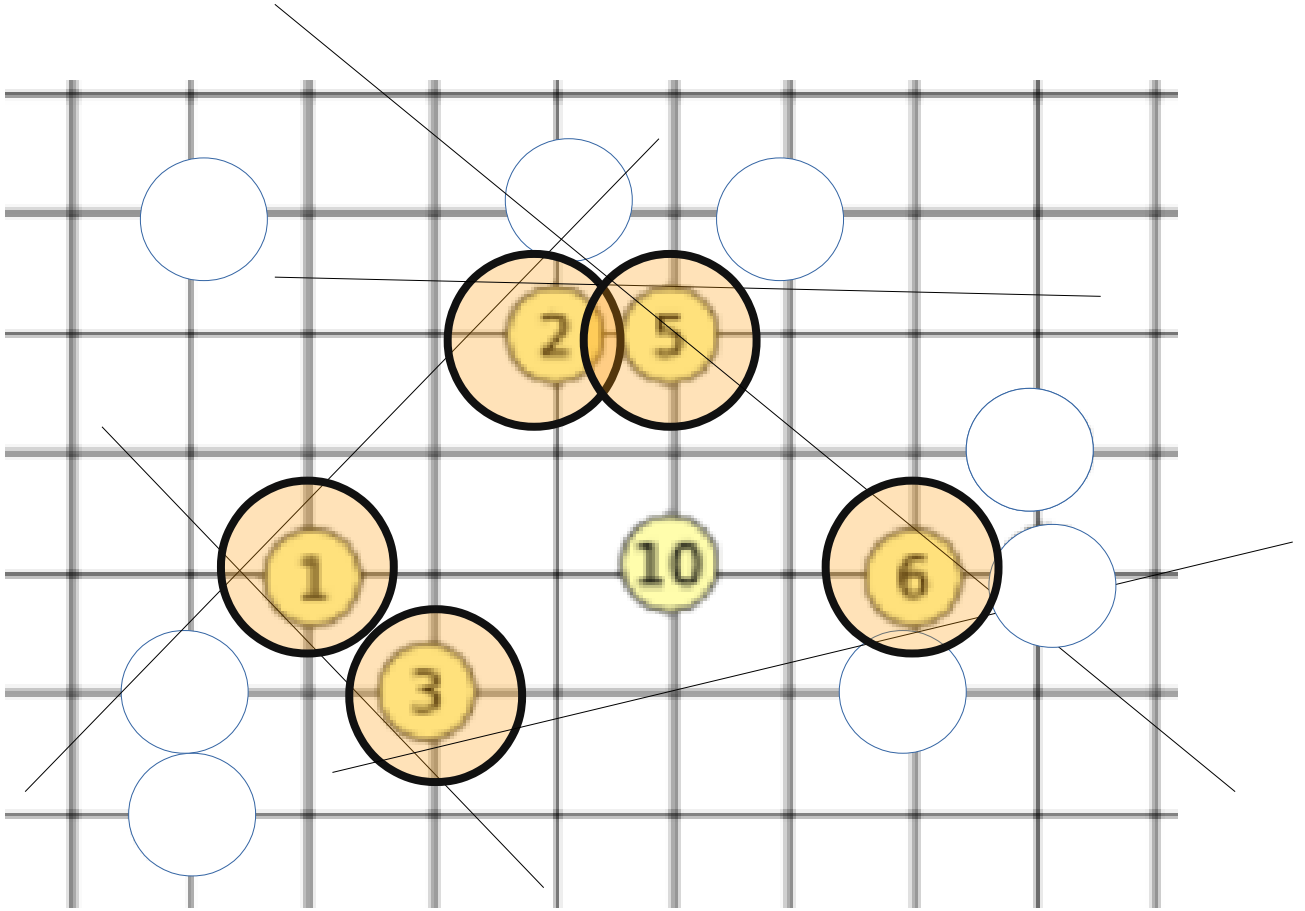
Depth 1 = { 12, 8, 14, 13, 4, 11, 9, 7 }



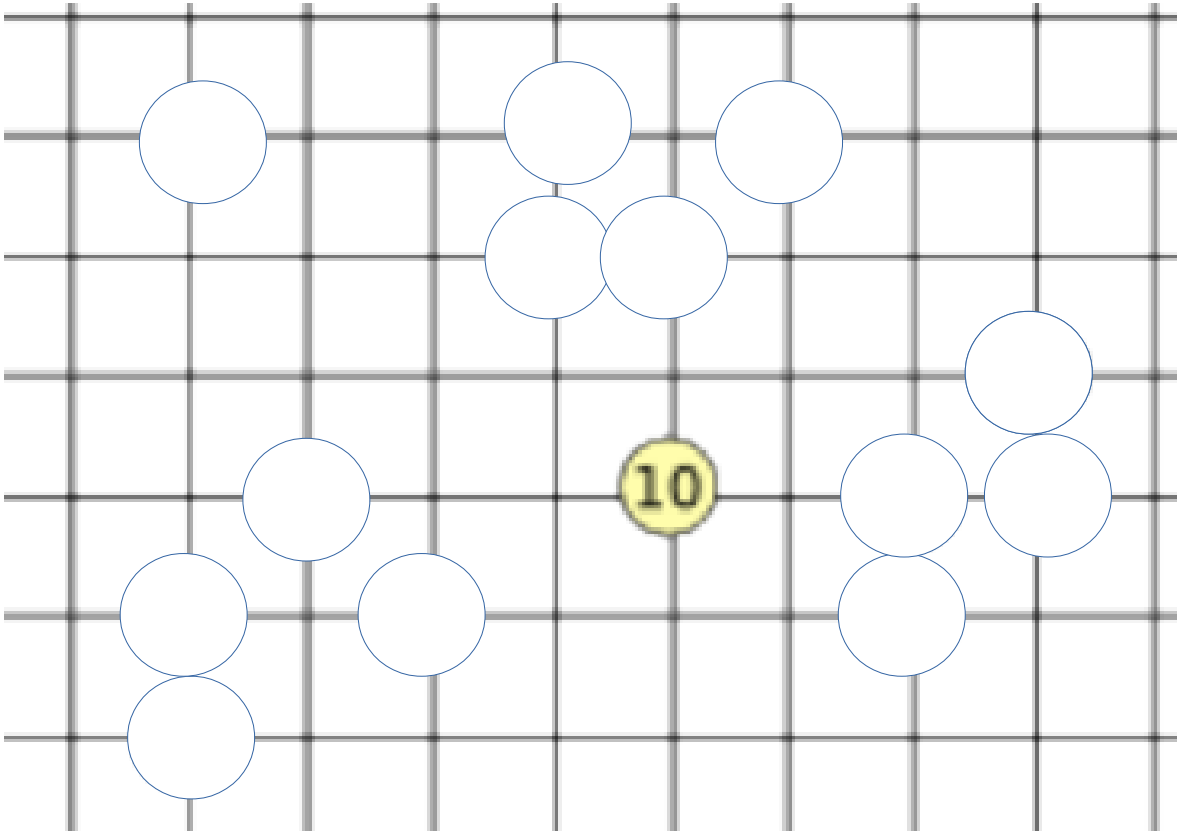
Depth 1 = { 12, 8, 14, 13, 4, 11, 9, 7 }



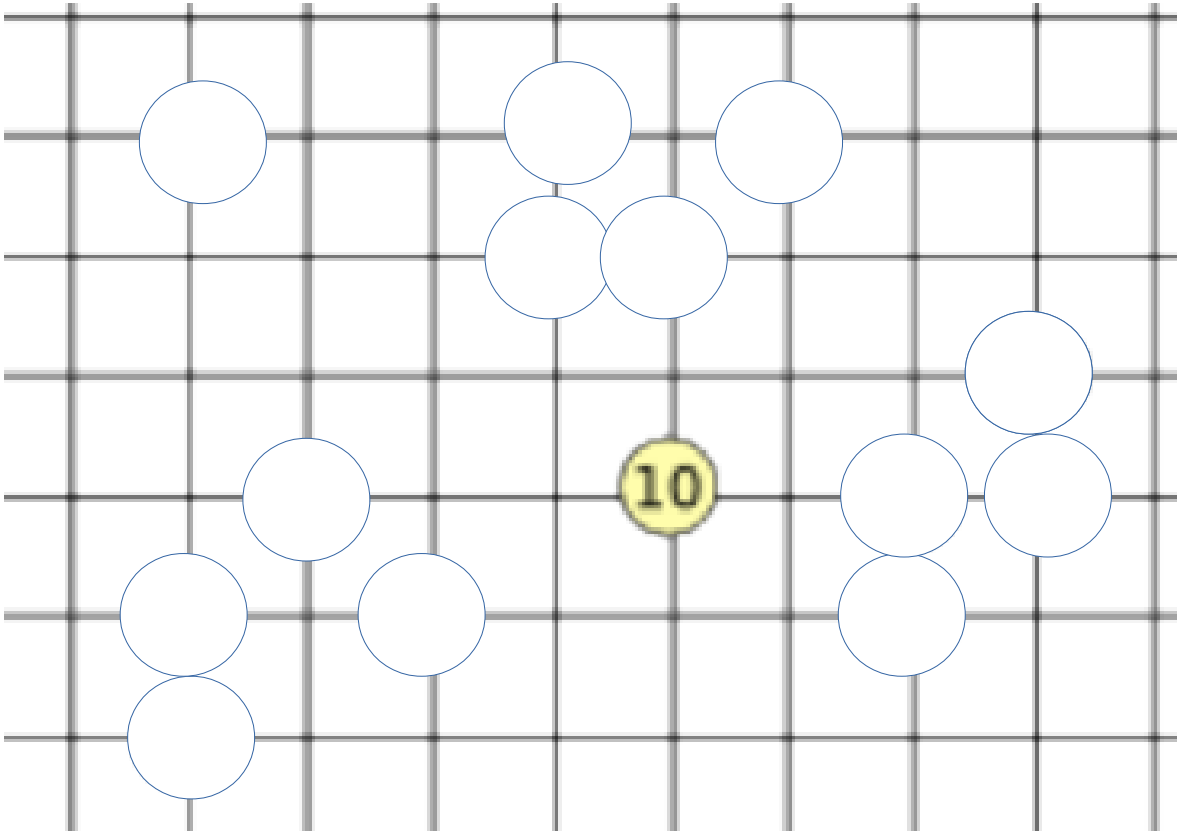
Depth 1 = { 12, 8, 14, 13, 4, 11, 9, 7 }
Depth 2 = { 2, 5, 6, 3, 1 }



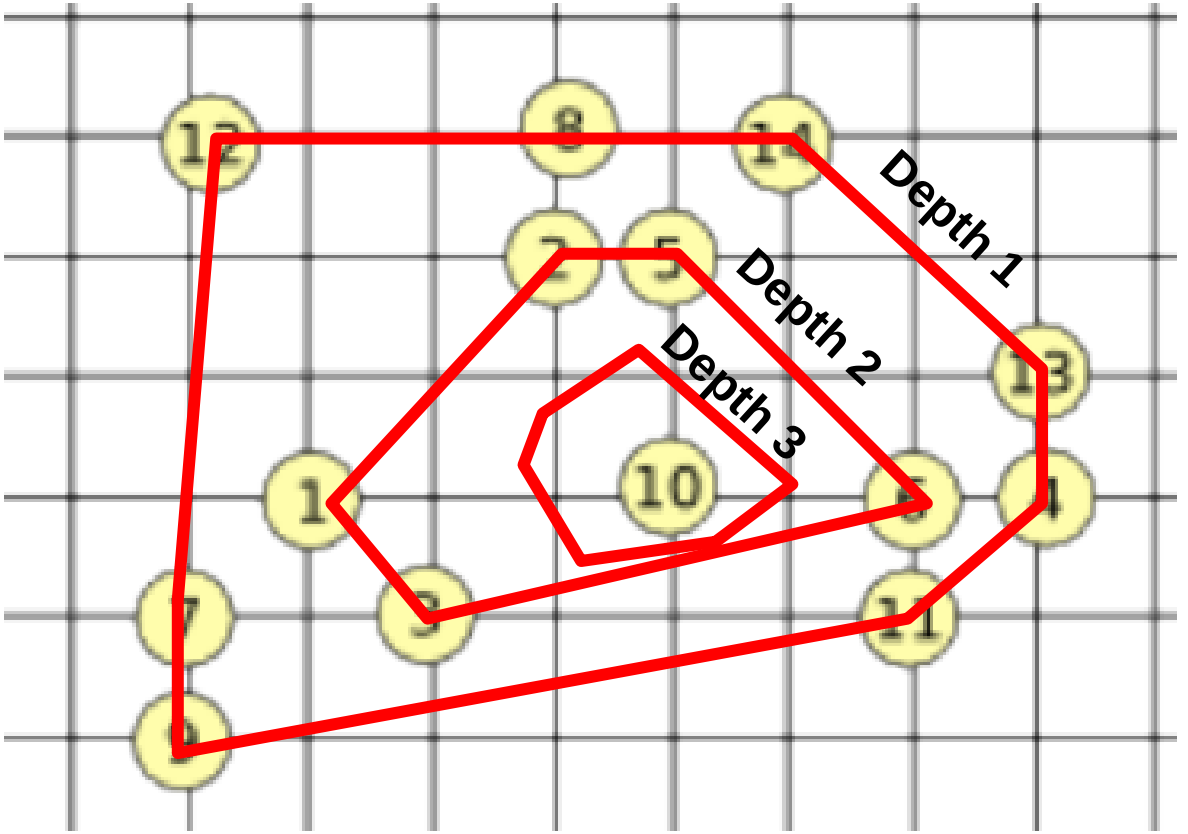
Depth 1 = { 12, 8, 14, 13, 4, 11, 9, 7 }
Depth 2 = { 2, 5, 6, 3, 1 }



Depth 1 = { 12, 8, 14, 13, 4, 11, 9, 7 }
Depth 2 = { 2, 5, 6, 3, 1 }
Depth 3 = { 10 }

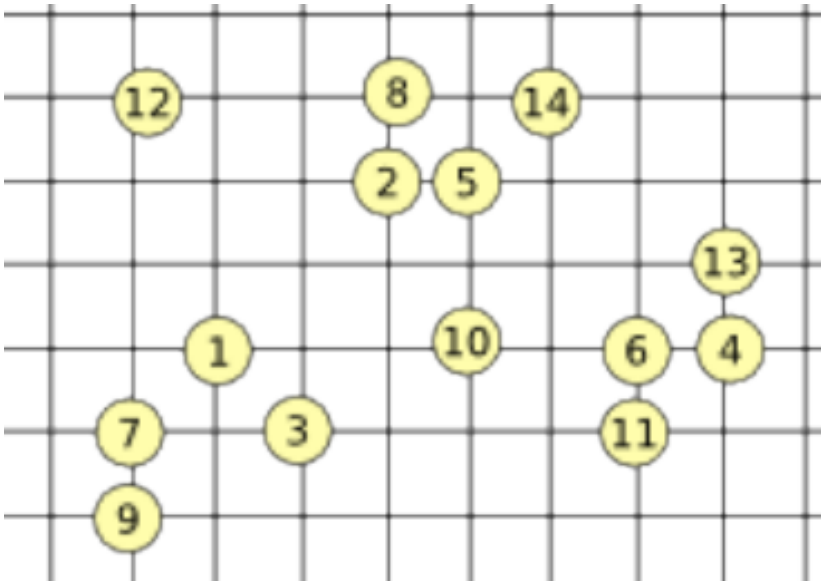


Depth 1 = { 12, 8, 14, 13, 4, 11, 9, 7 }
Depth 2 = { 2, 5, 6, 3, 1 }
Depth 3 = { 10 }



Angle-based outliers

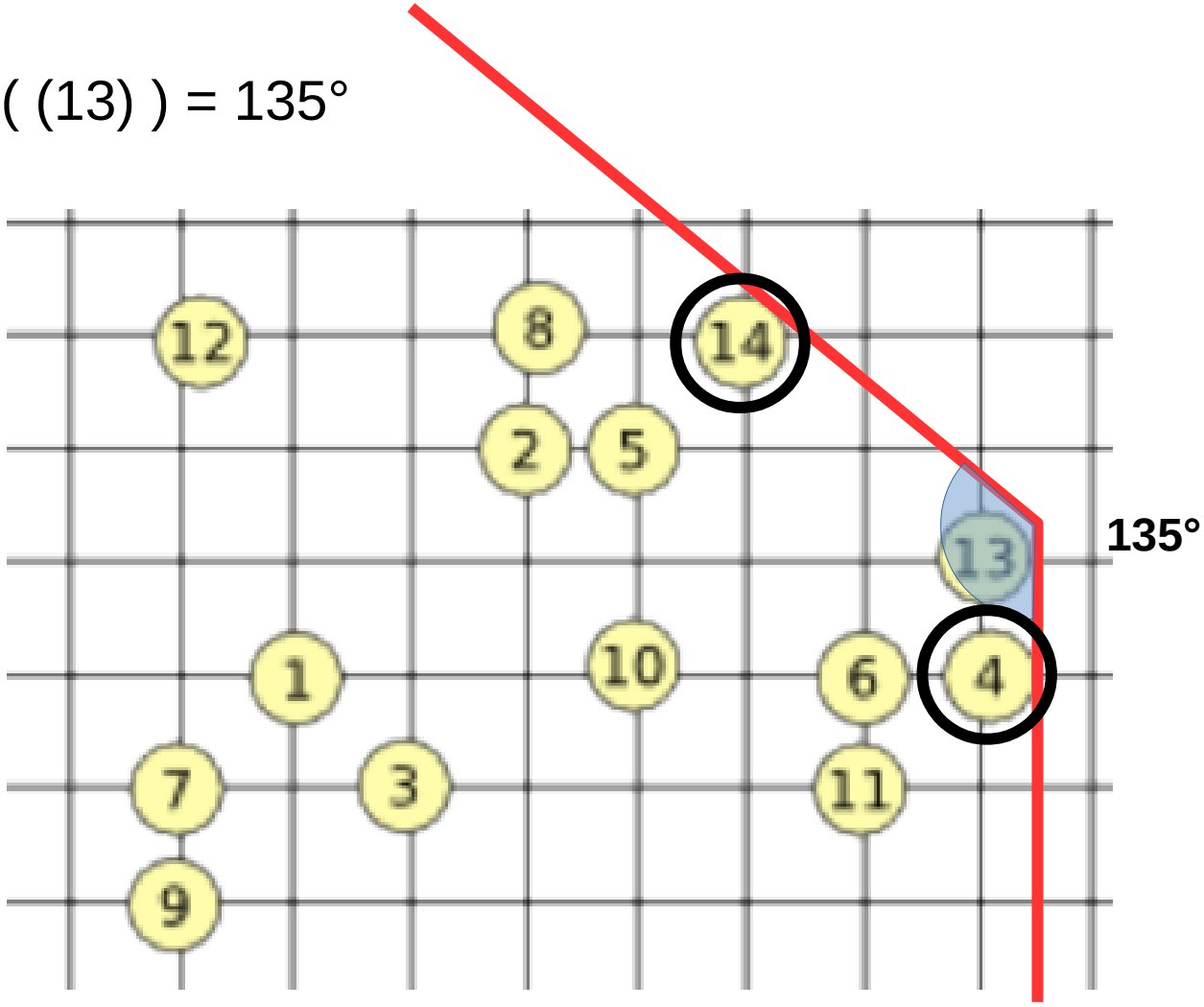
- Compute the simplified ABOD for each point
  - Simple-ABOD(p) = $MAX_{x,y \text{ in DB}}$ { angle($x\bar{p}y$) }
- Equivalent to find the smallest angle in p that contains the whole dataset



Angle-based outliers

Simple-ABOD(p) = $MAX_{x,y \text{ in } DB}$ { angle(x$\bar{p}$y) }

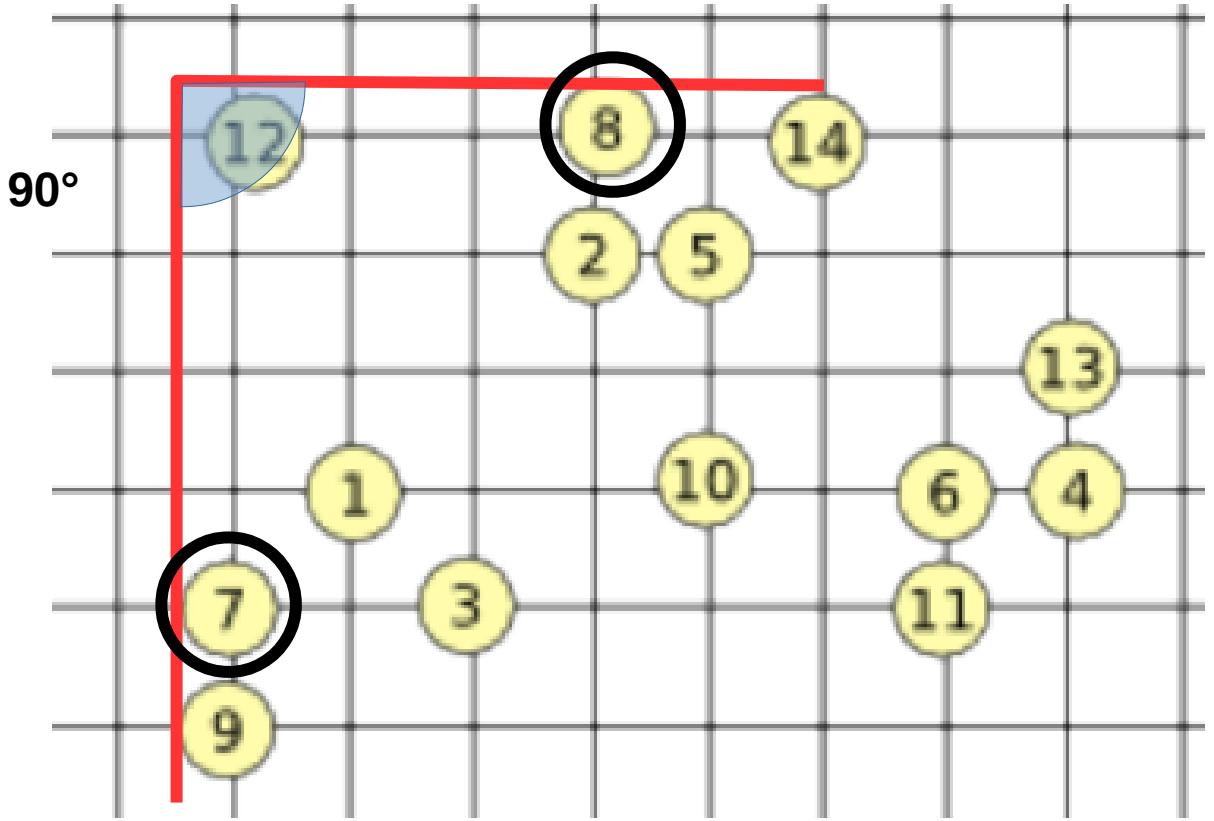
Simple-ABOD((13)) = 135°



Angle-based outliers

Simple-ABOD(p) = $MAX_{x,y \text{ in } DB}$ { angle($x\bar{\bar{p}}y$) }

Simple-ABOD((12)) = 90°



Angle-based outliers

Simple-ABOD(p) = $MAX_{x,y \text{ in } DB}$ { angle($x\bar{p}y$) }

Simple-ABOD((10)) = 180°