Coordinate transformations:

(x,y,xc,yc,alpha) => (xn,yn)

Shift:

* x0 = x – xc, y0 = y – yc

Alpha:

* xn = x0.cos(alpha) + y0.sin(alpha)
* yn = -x0.sin(alpha) + y0.cos(alpha)

The board plane: The zeppelin plane:

So add 180 to the given alpha

Now to get the x right, x = -x

Sorting according to polar angle:

0°: rightmost point, angles increase counterclockwise

Maybe switch this??

1. transform coordinates (around middle)
2. comparator:
   1. y1==0 && x1>0: P1 < P2 (P1:0°)
   2. y2==0 && x2>0: P1 > P2 (P2:0°)
   3. y1>0 && y2<0: P1<P2 (P1:[0,180],P2:[180,360])
   4. y2>0 && y1<0: P2<P1
   5. x1\*y2-y1\*x2 > 0 ==> P1<P2

Switch:

1. y1==0 && x1>0: P1 < P2 (P1:0°)
2. y2==0 && x2>0: P1 > P2 (P2:0°)
3. y1>0 && y2<0: P1>P2 (P1:[0,180],P2:[180,360])
4. y2>0 && y1<0: P2>P1
5. x1\*y2-y1\*x2 > 0 ==> P1>P2