

# Analysis: Blindfolded Bullseye

## Test Set 1

In Test Set 1 the radius is both really large and known in advance. The fact that it is large gives us a major restriction: plugging in the value for  $R$  in the limits for  $X$  and  $Y$  shows that the hidden center is restricted to a square with side length 10 nanometers, centered on the origin. That means that there are only  $11^2 = 121$  possible positions for the center! We can simply try throwing a dart at each of them, stopping as soon as we are told that we have hit the `CENTER`.

## Test Set 2

In Test Set 2 the radius is also large and known, but small enough that the square of candidate center points has side 101 nanometers. This means that there are 10201 possible centers, which is a lot more than the number of darts we get. We need to restrict that further.

We can use the fact that we know the exact radius to reduce this number. Every dart we throw can give us a clue. If a dart thrown at point  $p$  hits, that means the distance between  $p$  and the center  $c$  is no more than  $R$ . If a dart thrown at point  $q$  misses, that means that the distance between  $q$  and  $c$  is more than  $R$ . If we find two points  $p$  and  $q$  that are close to each other, the points that are no more than  $R$  away from  $p$  but more than  $R$  away from  $q$  form a narrow crescent-shaped region of width equal to the distance between  $p$  and  $q$ . Intersecting that with the original square of possibilities from the first paragraph can give us a small enough range for  $c$  that we can just throw a dart at each possibility without running out of darts.

We can find points  $p$  and  $q$  that are close to each other by finding the edge of the dartboard. Since the dartboard is so large, the edge has to be close to the edge of the wall. If we inspect points  $(x, 0)$  for the possible values of  $x$ , most points are guaranteed to be inside the dartboard: any point  $(x, 0)$  with  $-10^9 + 101 \leq x \leq 10^9 - 101$  is no more than  $R$  away from all possible centers, and is thus guaranteed to be inside the circle. Therefore, we can try all possible  $x$  on any one side of that guaranteed interval to find a hit point and a miss point that are 1 nanometer away from each other. Since there are only 101 points on each side that are not guaranteed to be in the circle — for example  $[-10^9, -10^9 + 100]$  — this requires at most 101 darts.

After finding those points, we know that the center has to be inside a narrow crescent-shaped band of width 1. Intersecting that with the 101-by-101 square of candidates leaves at most 202 candidates, since there can't be more than 2 candidates that share an  $x$  value. Moreover, for most  $x$  values, the rounding ensures that there are less than 2, so the 199 darts we have left are enough to cover all candidates.

Notice that no complicated geometry is involved to find the candidates; the square is small enough to iterate over all points within it and check the distances to our hit and miss points to see if they are candidates or not.

## Test Set 3

For Test Set 3, we can take a more typical geometric approach. To find the center of a circle, we can find 3 points on the edge of the circle, and then calculate a center from those. This has an issue: we only have nanometer precision, which means we would typically find points near the edge of the circle, but not exactly in it. The error that this introduces in our calculations could be significant. We can overcome this by bounding the error and then checking not only our

calculated center, but also other nearby points too. The math to calculate the center, and most importantly, to bound the error of it, can be hard and time consuming, but it is doable. In addition, you can wing it by not bounding the error and instead starting to throw at your calculated center and then spiral around it on nearby points until you find the actual center. This requires a leap of faith that you won't run out of tries, but it's a reasonable assumption. Fortunately, there is a simpler related approach that is more clearly correct, that we describe below. Of course, we still need to solve the problem of finding "almost-edge" points of the circle, which we also need for our simpler approach.

To find a point in the edge, we can't use the approach in Test Set 1, because if the dartboard's radius is not very large, those points could be really far away from the wall's edge. A different way of doing it is via a [binary search](#). Say we know that point  $(x_0, y_0)$  is within the dartboard.

Then, we know that for all  $x$  in the range  $[-10^9, x_0]$  the points  $(x, y_0)$  are grouped such that there is a (possibly empty) interval having all misses, and then an interval having all hits. The same holds (in reverse) for  $[x_0, 10^9]$ . Analogously, points  $(x_0, y)$  for  $y$  in  $[-10^9, y_0]$  or  $[y_0, 10^9]$  are grouped by hits/misses. Then, for each of those ranges, we can do a binary search to find the switching point — that is, a point in the edge of the circle. This is exactly what we did in Test Set 2, except we fixed  $x_0 = 0$ , because the radius is so large that some points like  $(0, 0)$  are guaranteed to be inside the dartboard, and the range is small enough that we can scan it entirely instead of using binary search.

The searches above get us leftmost and rightmost points inside the dartboard for a given  $y$ -coordinate  $y_0$ , and topmost and bottommost points for a given  $x$ -coordinate  $x_0$ . Notice that the dartboard is symmetric with respect to the horizontal and vertical lines that go through its center. Therefore, the leftmost and rightmost points inside the dartboard in any fixed  $y$ -coordinate mirror each other, and the  $x$ -coordinate of the center is the midpoint between their  $x$ -coordinates. Similarly, we can find the  $y$ -coordinate of the center as the midpoint between the  $y$ -coordinates of the topmost and bottommost points inside the dartboard at any fixed  $x$ -coordinate.

The remaining task is to find a single point inside the dartboard. Since the area of the dartboard makes up a significant percentage of the wall area (at least  $\pi / 16$ ), we could do this by throwing darts at random until we hit one point (the probability of hitting could be slightly less than  $\pi / 16$  because we consider only points with integer nanometer coordinates, but the difference is negligible). This has an overwhelmingly high chance of hitting in 100 throws or fewer for all cases. However, there is a deterministic way that also works, which is to divide the wall into squares of side **A**. If the center of the dartboard is inside a square, it is guaranteed that at least one of the corners of that square is inside the dartboard. Therefore, we can simply try all those corners (there are only 25, and it is even possible to restrict it further), and we will find a hit point.

So, we need a fixed small number of darts (up to 100, depending on our method of choice) to find a point inside the dartboard, and each binary search needs at most  $31 = \text{ceil}(\log_2 (2 \times 10^9 + 1))$ . That is at most  $4 \times 31 + 100$ , which is a lot less than the 300 limit. In the first version, you need one fewer binary search — 3 edge points are enough to find a unique center — so after finding the center with error, you could have 150 darts left to explore its vicinity. Notice that if your initial point is actually one of the points  $(X - R, Y)$ ,  $(X + R, Y)$ ,  $(X, Y - R)$  or  $(X, Y + R)$ , you would end up with two points instead of three and be unable to find the center. If we use the random procedure to find  $(x_0, y_0)$ , the probability of this happening is negligible. Otherwise, we can detect the case and know that the two found points are opposite each other, so the center is their middle point.