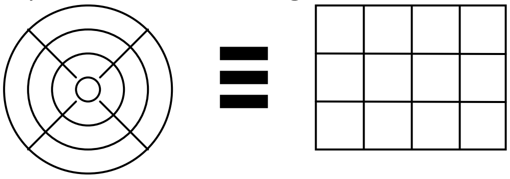
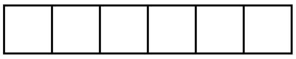
Modular Knights

The first thing I noticed about this problem is that the circular chess board is equivalent to a rectangular chess board that loops back onto itself:



So if you start on the top sector of the circular board then this is equivalent to starting on the top left square of the rectangular board. You can test this by considering the move 3 down and 4 along – you get back to where you started in both. Now let's start labelling things. Let's call the number of sectors ***p*** and the number of tracks ***q***. This means that the rectangular board will have ***p*** rows and ***q*** columns and I will call it a ***p***x***q*** board. The movement of the knight can now be denoted as ***a*** spaces in 1 direction and ***b*** in the other. Now we will work with a single track and see what we can deduce from it:



This is our test track and as you can see it is a 1x6 board. Imagine that our knight starts on the leftmost square and moves 7 to the right. This is equivalent to moving 1 to the right, or moving 13 to the right, or moving 5 to the left. Now to spot the pattern here you must imagine that the direction the knight is moving can be positive or negative, like displacement in physics, and I will take the direction to the right as being positive and the left as being negative. I will also now write "move right 7 spaces" as "move +7 spaces", so (+7 spaces) = (+1 spaces) = (-5 spaces). You may have noticed that each value is 6 away from the next one, and therefore the remainder is the same in each case if you divided them by 6. To show this we can apply the mod operator:

+7 (mod 6) = +1 (mod 6) (7-1x6)
+13 (mod 6) = +1 (mod 6) (13-2x6)
-5 (mod 6) = +1 (mod 6) (-5+1x6)

Therefore they are all equivalent to moving +1 space, and we will call this smallest positive value the relative number of moves. This will help us with the next bit. On our test track you can see that it can step on every space if it moves +1 or +5 (mod 6). My explanation to this problem is that if you have the 1x6 grid and move +3 spaces each time you won't be able to reach every square because 6 is a multiple of 3, and the same reasoning applies to +2 spaces as well. But as you may have noticed this reasoning doesn't work for 4 because 6 isn't a multiple of 4. If you draw a table of the first couple of multiples of each of the numbers up to and including 6 you'll see that the moves that can get to all the squares only share a common multiple when they are multiplied together, whereas the ones that can't make it to all of the spaces have a common multiple before this. My refined model is therefore that the knight can't make it to every space in a line if the relative number of moves along one axis and the distance the knight can move in the same direction share a common prime factor board. In other words that's:

The knight can jump to every space in a straight line if ***a*** (mod ***p***) and ***p*** don't have a common prime factor.
If you apply this reasoning into 2 dimensions it gets a bit harder to work out, but if you assume that the knight can only move n spaces in one direction and m spaces in the other then you get an odd looking logic equation thing – to make it more mathematical and less wordy "a?b" means "do a and b not share a common prime factor" and is true if they don't share a prime factor and false if they do. The logic equation thing looks like this:
A: (***a*** (mod ***p***))?***p***
B: (***b*** (mod ***q***))?***q***
C: (***b*** (mod ***p***))?***p***
D: (***a*** (mod ***q***))?***q***
Then the knight can certainly jump to each space if (A and B) are true, or if (C and D) are true. To prove this I looked for the highest settings for knights moves (***a***=6, ***b***=4) and then found the track size 11x7 solves the logic equation and therefore there must be a solution which I then found:

This proves it in a way because the idea it was derived from was correct and 1 correct example therefore proves it. But that can't be the whole equation because there is a solution to ***a***=2, ***b***=3 on a 6x3 board.

This combination makes A and C false and therefore a solution shouldn't exist, but it does. After some more thought I managed to derive another equation. To explain how I derived it we first remove the assumption that a is in the same direction as p and the same with b and q or the other way round and think about the combinations that could exist (ie changing direction each go). This then leads to another variation that we need to add to the equation – "((A or B) and (C or D))". This means that the equation becomes:
A = (***a*** (mod ***p***))?***p***
B = (***b*** (mod ***q***))?***q***

C = (***b*** (mod ***p***))?***p***
D = (***a*** (mod ***q***))?***q***
Then the knight can certainly jump to each space if (A and B) or (C and D) or ((A or B) and (C or D)) are true.
Now this looks very complicated, but by looking at how the "or" is like addition and the "and" is like multiplication you can expand this out:
(A or B) and C = (A and C) or (B and C)
(A and B) or C cannot be simplified
The logic equation then becomes:
((A + B) x (C + D)) + (A x B) + (C x D)
= A x (C + D) + B x (C + D) + (A x B) + (C x D)
= (A x C) + (A x D) + (B x C) + (B x D) + (A x B) + (C x D)
= (A and B) or (A and C) or (A and D) or (B and C) or (B and D) or (C and D)
Which you can see is every possible combination of A, B, C and D. This means that as long as 2 of the logic statements are true then there is a solution, but there's no way of helping find that solution. For example I can tell you there's a solution to a 182x97 board with ***a***=31, ***b***=25 but I have no idea how you go about finding it. This also shows that a 6x3 board with ***a***=2, ***b***=3 is possible.
Therefore the final equation is:
A = (***a*** (mod ***p***))?***p***
B = (***b*** (mod ***q***))?***q***
C = (***b*** (mod ***p***))?***p***
D = (***a*** (mod ***q***))?***q***
Then the knight can certainly jump to each space as long as at least 2 of these are true.