

## [Investigating pi with Scratch]

We can use scratch to investigate pi. This gives us an idea of how algorithms and software can be used to find estimates or approximations of a problem's solution. It also emphasises the meaning of pi as the ratio between a circle's circumference and its diameter.

Our method of approach will be to draw polygons and then use the fact that scratch can sense the coordinates of the cat's position in order to calculate how far away the cat is once it's half way around the polygon.

The more sides the polygon has, the closer it will be, relatively, to a circle and then so we'll be calculating the ratio of the circumference, which is just the sum of how many sides there are, to the distance.

Then for the distance, we can use pythagoras's theorem, where the sides of the triangle are just the x and y coordinates.

[in scratch]

The only input we'll need is the number of sides. The idea is that the higher the number of sides, the closer it looks to a circle and the more accurate the approximation of pi will be.

After we know n, we can draw the polygon. Now other times we did this, we made the sides big so that we could see them. In this case, we'll set the step size to 1 so that the distance around the polygon is just n. The amount that we turn is  $360/n$ .

Now rather than draw a whole polygon, we'll just draw half a polygon, so that at the end, we can read off the x and y coordinates and see how far away the cat has moved.

We'll set the starting points to 0, and erase the page when we start to keep it looking clean.

Now the diameter calculation. We're using Pythagoras's theorem for this, the square root of the x coordinate squared plus the y coordinate squared. There doesn't seem to be a squared operation in scratch, so we'll just use multiply, x by x and y by y.

Now when we run the program and enter different values, we can see that the number is smaller than the actual value of pi, because the polygons take a shorter path around them than the circle, but we get closer and closer to pi.

We just need to be careful that the cat doesn't go too far outside the window, as this means the program won't adjust its position coordinates properly.