

archimedes_454_simulation

January 18, 2020

```
[ ]: # Code to run Archimedes' Method of Exhaustion to approximate the value of pi.  
# Author: Michael Kumaresan  
  
import numpy as np  
import turtle  
import math  
from math import sqrt  
from turtle import Turtle, Screen  
  
SCALE=300  
ITERATIONS=7  
  
def draw_circle(turtle):  
    turtle.goto(0, -SCALE)  
    turtle.pendown()  
    turtle.circle(SCALE)  
    turtle.penup()  
  
def inscribe_circle(turtle, sides, edge_length):  
    turtle.goto(0, -SCALE)  
    turtle.setheading(0)  
    turtle.left(180 / sides)  
    turtle.pendown()  
    for _ in range(sides):  
        turtle.forward(edge_length * SCALE)  
        turtle.left(360 / sides)  
    turtle.penup()  
  
def pi_archimedes(turtle, n):  
    polygon_edge_length_squared = 2.0  
    polygon_edge_length = sqrt(polygon_edge_length_squared)  
    polygon_sides = 4  
  
    inscribe_circle(turtle, polygon_sides, polygon_edge_length)  
    print(polygon_sides * polygon_edge_length / 2)
```

```

    for i in range(n-1):
        polygon_sides *=2
        polygon_edge_length_squared = 2 - 2 *  $\cos$ 
    →sqrt(1-polygon_edge_length_squared / 4)
        polygon_edge_length = sqrt(polygon_edge_length_squared)
        inscribe_circle(turtle, polygon_sides, polygon_edge_length)
        print(polygon_sides * polygon_edge_length / 2)

yertle = Turtle()
yertle.penup()
draw_circle(yertle)
pi_archimedes(yertle, ITERATIONS)
yertle.hideturtle()
Screen().exitonclick()

```