INSCRIBED FIGURES

P.Sravanthi: 21B01A5486: AIDS-B

Sk.Kareena: 21BO1A0249: EEE

B.kalyani: 21B01A1218: IT-A

T.Ekeswari: 21B01A12I2: IT-C

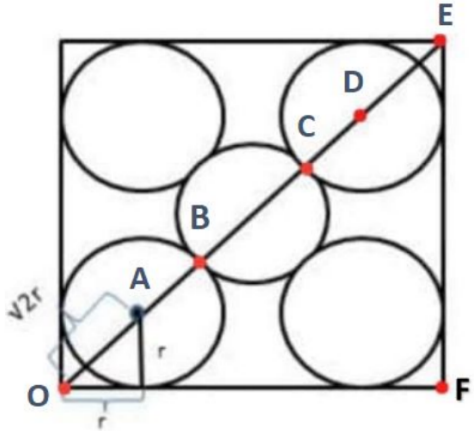
S.Bhanu Sri: 22B05A1216: IT-C

MARCH 25, 2023

INTRODUCTION:

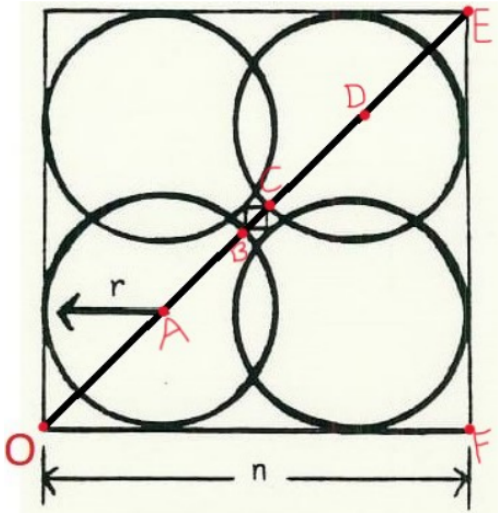
- Given a square of length 'n' and four circles of equal radius 'r' inscribed in it where each circle is located in each corner and tangent to two sides of the square. we have to calculate the area of the largest circle or square that lies completely within the large square and intersects all the four circles in atmost four points.

If the shape is circle:



- The distance from the corner of square to centre of circle,
  OA = DE = 2 r
  By Pythagoras theorem,
  $OE^2 = OF^2 + EF^2$
  $OE = (l^2 + l^2)$
  OA + AB + BC + CD + DE = 2l
  2r + r + x + r + 2r = 2l
  x + 2r + 22r = 2l
  x + 2r(1+ 2) = 2l
  x = 2l - 2r(1+ 2)
  Here x is the diameter of the required circle
  Area of the required circle,
  = 3.14 × (x/2) × (x/2)

If the shape is square:



- The distance from the corner of square to centre of circle, x=2 r
  By Pythagoras theorem,
  $OE^2 = OF^2 + EF^2$
  $OE = (l^2 + l^2)$
  OA + AB + BC + CD + DE = 2l
  2r + r + x + r + 2r = 2l
  x + 2r + 22r = 2l
  x + 2r(1+ 2) = 2l
  x = 2l - 2r(1+ 2)
  Here x is the length of diagonal of the required square
  Area of the required square = (x / $\sqrt{2}$)$(x/\sqrt{2})$

PACKAGES:

- Math Module: we have imported math module in this project. The math module is a built in module that contains a set of math methods and constants. we have used the math method "math.sqrt()" to return the square root of a number and the math constant "math.pi" to return the value of pi.
- Sys Module: we have imported sys module and used it's variable "sys.argv" which is a list of command line arguments. In this project, length, radius and shape are passed as command line arguments.

CHALLENGES:

- If the given diameter of the circle exceeds the given length of the large square, then all the circles get overlapped and there will be no inscribed circle or square that intersects all the four circles in atmost four points.
- To overcome this, we have given the condition that the diameter of the circle should be less than the length of the large square

STATISTICS:

- Number of lines of code : 24
- Number of functions : 3

  areaOfCircle( )
  areaOfSquare( )
  inscribedfigures( )

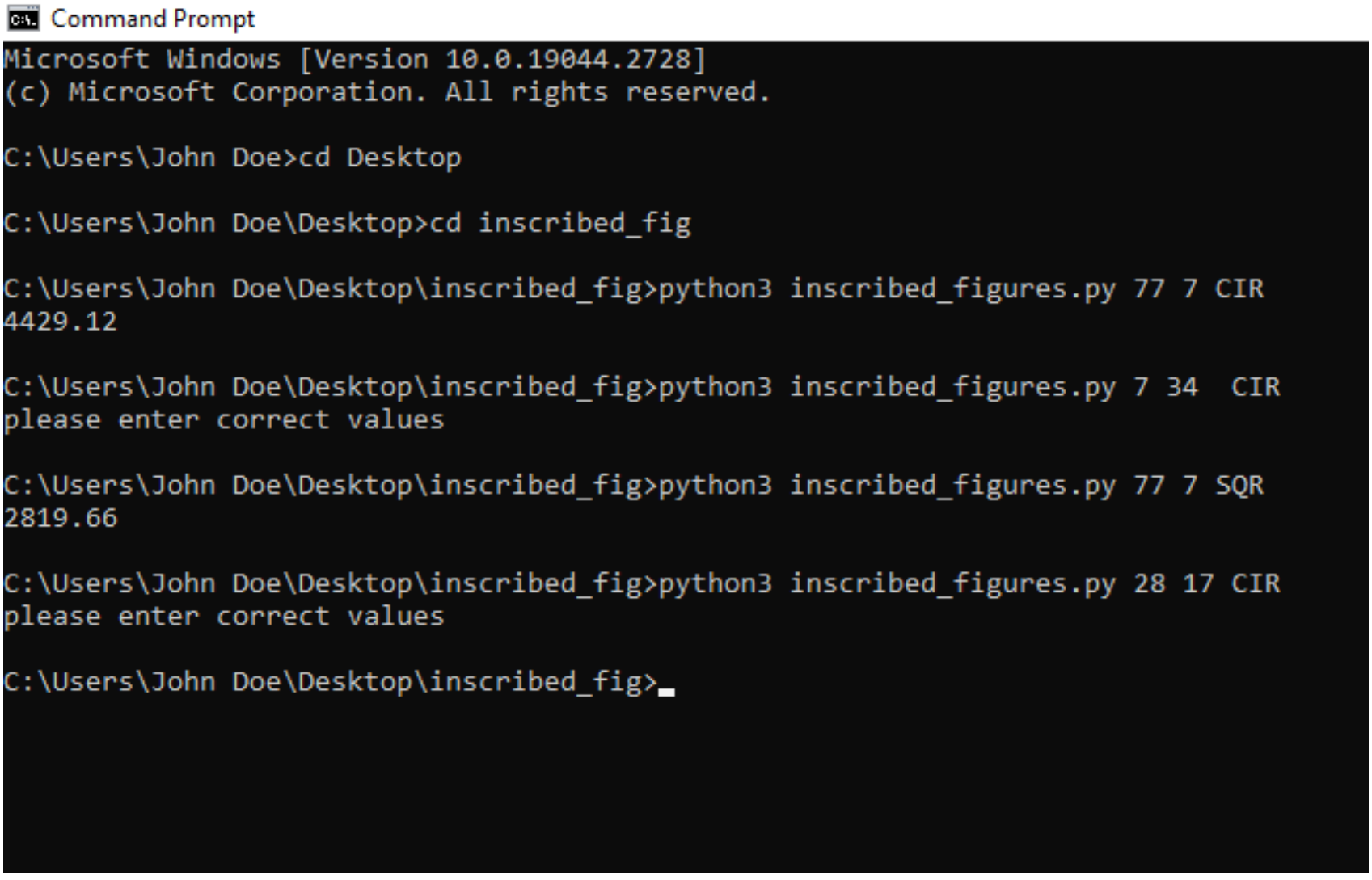
PROGRAM:

inscribed_figures.py

C: > Users > John Doe > Desktop > inscribed_fig > inscribed_figures.py

```python
import math
import sys

def areaOfcircle(length,radius):
    diagofreqcir = math.sqrt(2) * length-(2 * (math.sqrt(2) * radius))-(2 * radius)
    return math.pi * diagofreqcir / 2 * diagofreqcir / 2


def areaofsquare(length,radius):
    diagofreqsqr = math.sqrt(2) * length - (2 * (math.sqrt(2) * radius))-(2 * radius)
    return diagofreqsqr / math.sqrt(2) * diagofreqsqr / math.sqrt(2)

def inscribed_figures(length,radius,shape):
    if shape == 'CIR' and 2 * radius < length:
        return f"{areaOfcircle(length,radius):.2f}"
    elif shape == 'SQR' and 2 * radius < length:
        return f"{areaofsquare(length,radius):.2f}"
    else:
        return "please enter correct values"

print(inscribed_figures(int(sys.argv[1]),int(sys.argv[2]),sys.argv[3]))

```

OUTPUT:


Command Prompt
Microsoft Windows [Version 10.0.19044.2728]
(c) Microsoft Corporation. All rights reserved.
C:\Users\John Doe>cd Desktop
C:\Users\John Doe\Desktop>cd inscribed_fig
C:\Users\John Doe\Desktop\inscribed_fig>python3 inscribed_figures.py 77 7 CIR
4429.12
C:\Users\John Doe\Desktop\inscribed_fig>python3 inscribed_figures.py 7 34 CIR
please enter correct values
C:\Users\John Doe\Desktop\inscribed_fig>python3 inscribed_figures.py 77 7 SQR
2819.66
C:\Users\John Doe\Desktop\inscribed_fig>python3 inscribed_figures.py 28 17 CIR
please enter correct values
C:\Users\John Doe\Desktop\inscribed_fig>

Thank you!