INSCRIBED FIGURES

PRIYANKA IPPALA : 21B01A1259 :IT-A

GUNDABATTULA MADHURI : 21BO1A1258: IT-A

PERURI SRI VEDHA : 21B01A12E1 : IT-C

DAMMALA MADHUNIKA : 21B01A0206 : EEE

YANDRA PURNA VIJAYA KUMARI : 21B01A54C7 : AIDS-B

MARCH 25, 2023

Contents

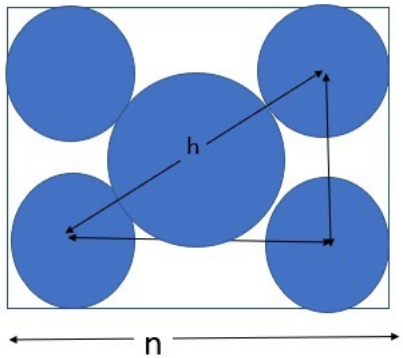
INTRODUCTION

INTRODUCTION:

- Given a square of length 'n' and four equal circles of 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.

APPROACH FOR CIRCLE

If the shape is circle:



n = length of side of square

r = radius of circles

R = Radius of inscribed circle

Area = Area of circle inscribed

Derived:

Side = side of triangle = (n – 2*r)

h = hypotenuse of triangle

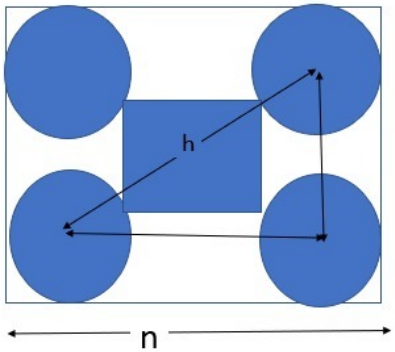
By Pythagoras Theorem:

h = sqrt(side**2)

R = (h – (2 * r)) / 2
Area = (pi * (R ** 2))

APPROACH FOR SQUARE

If the shape is square:



n = length of side of square

r = radius of circles

D = diagonal length of new square inscribed

Area = Area of inscribed square

Derived:

Side = side of triangle = (n – 2*r)

h = hypotenuse of triangle

By Pythagoras Theorem:

h = sqrt(side**2)

D = (h - (2*r))

Area = (D ** 2) / 2

LEARNINGS

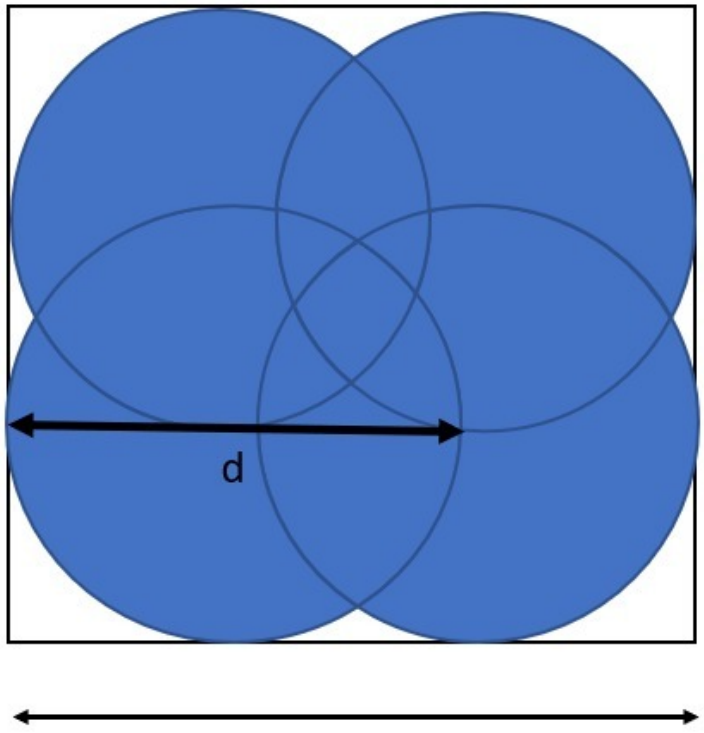
PACKAGES:

- 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

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



n = side of square

d = diagonal of circle

STATISTICS

STATISTICS:

- Number of lines of code : 35
- Number of functions : 4

  diagonal( )
  findarea( )
  areaCircle( )
  squarearea( )

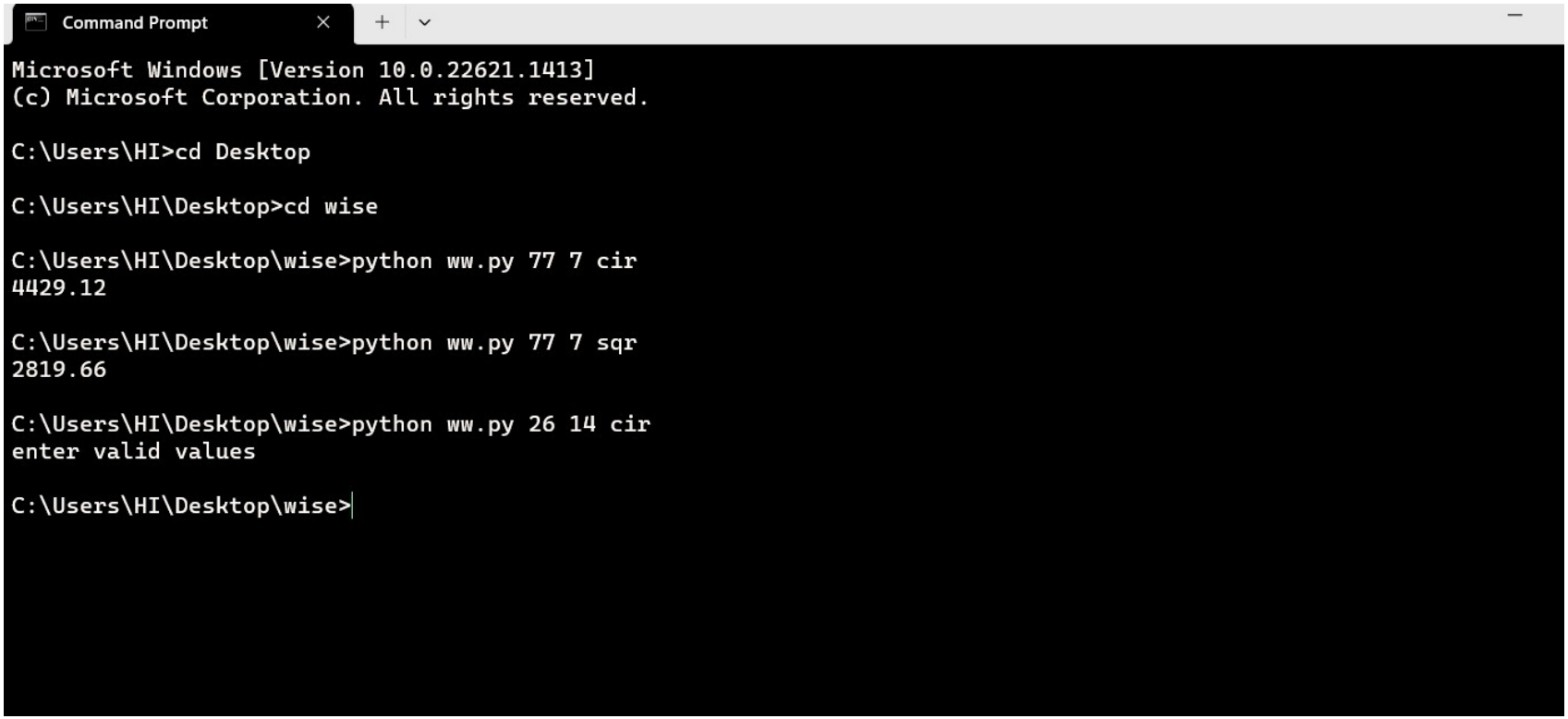
DEMO

PROGRAM:

ww.py

```python
import math
import sys
def areaCircle(diagonal):
    radius = diagonal/2
    return (f'{math.pi * (radius**2):.2f}')

def squarearea(diameter):
    side = diameter / math.sqrt(2)
    area = side*side
    return (f'{area:.2f}')

def findarea(diagonal,shape):
    if diagonal>=0:
        if shape=='cir':
            area = areaCircle(diagonal)
        elif shape== 'sqr':
            area = squarearea(diagonal)
        else:
            print('enter valid shape')
    else:
        area = 'enter valid values'
    return area

def diagonal(n,r):
    t_side = n-2*r
    t_hypo = math.sqrt(2)*(t_side)
    dia = t_hypo-2*r
    return dia

n = int(sys.argv[1])
r = int(sys.argv[2])
shape = sys.argv[3]
dia = diagonal(n,r)
area = findarea(dia,shape)
print(area)
```

OUTPUT:


Command Prompt
Microsoft Windows [Version 10.0.22621.1413]
(c) Microsoft Corporation. All rights reserved.
C:\Users\HI>cd Desktop
C:\Users\HI\Desktop>cd wise
C:\Users\HI\Desktop\wise>python ww.py 77 7 cir
4429.12
C:\Users\HI\Desktop\wise>python ww.py 77 7 sqr
2819.66
C:\Users\HI\Desktop\wise>python ww.py 26 14 cir
enter valid values
C:\Users\HI\Desktop\wise>

Thank you!