Tail Nesting

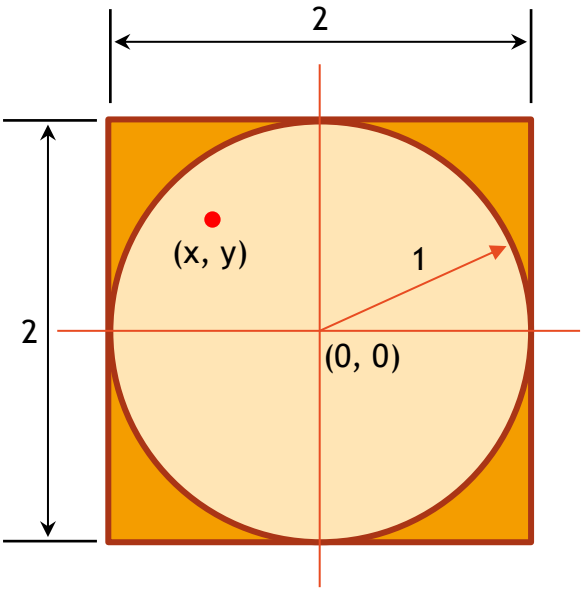
All nested selections occur in the else

02-07-tail-nesting-grades.py

```
score = int(input('Enter score: '))
if score >= 90:
    grade = 'A'
else:
    if score >= 80:
        grade = 'B'
    else:
        if score >= 70:
            grade = 'C'
        else:
            if score >= 60:
                grade = 'D'
            else:
                grade = 'F'

print(f'Grade is {grade}')
```

```
score = int(input('Enter score: '))

if score >= 90:
    grade = 'A'
elif score >= 80:
    grade = 'B'
elif score >= 70:
    grade = 'C'
elif score >= 60:
    grade = 'D'
else:
    grade = 'F'

print(f'Grade is {grade}')
```

Back to the simulation



$$Square\ Area = 2 \times 2 = 4$$

$$\text{Circle Area} = \pi r^2 = \pi \times 1^2 = \pi$$

$$\frac{Circle\ Area}{Square\ Area} = \frac{\pi}{4}$$

$$\pi = 4 \cdot \frac{Circle\ Area}{Square\ Area}$$

How to "throw" a Dart

- Random x: -1 to +1
- Random y: -1 to +1

How to tell if a dart is inside the circle

1. Can use distance formula

$$dist = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Since $(x_1, y_1) = (0, 0)$

$$dist = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

$$= \sqrt{(x_2 - 0)^2 + (y_2 - 0)^2}$$

$$= \sqrt{{x_2}^2 + {y_2}^2}$$

Just call the point (x, y) $dist = \sqrt{x^2 + y^2}$

Don't need the distance - Just need to know if it's ≥ 1

$\sqrt{x^2+y^2} > 1$	$x^2+y^2 > 1$
$\sqrt{x^2+y^2} = 1$	$x^2+y^2 = 1$
$\sqrt{x^2+y^2} < 1$	$x^2+y^2 < 1$

In other words, $\sqrt{x^2+y^2} \leq 1 \iff x^2+y^2 \leq 1$

So, a dart at (x, y) is in the circle if $x^2+y^2 \leq 1$

Write the Function

See 02-08-MontePi.py

```
def montePi(numDarts):
    numInCircle = 0

    for i in range(numDarts):
        # "Throw" the dart
        x = random.uniform(-1, 1)
        y = random.uniform(-1, 1)

        # Count it if it landed in the circle
        if x*x + y*y <= 1:
            numInCircle += 1

    piApprox = (numInCircle / numDarts) * 4
    return piApprox
```

Animate the Simulation

Set World Coordinates

- Want the unit circle to be big on the canvas
  - turtle.setworldcoordinates(LLx, LLy, URx, URy)

bv

(URx, URy)

(LLx, LLy)

See 02-09-monteDraw.py