

Laço de repetição

Guilherme Dal Bianco

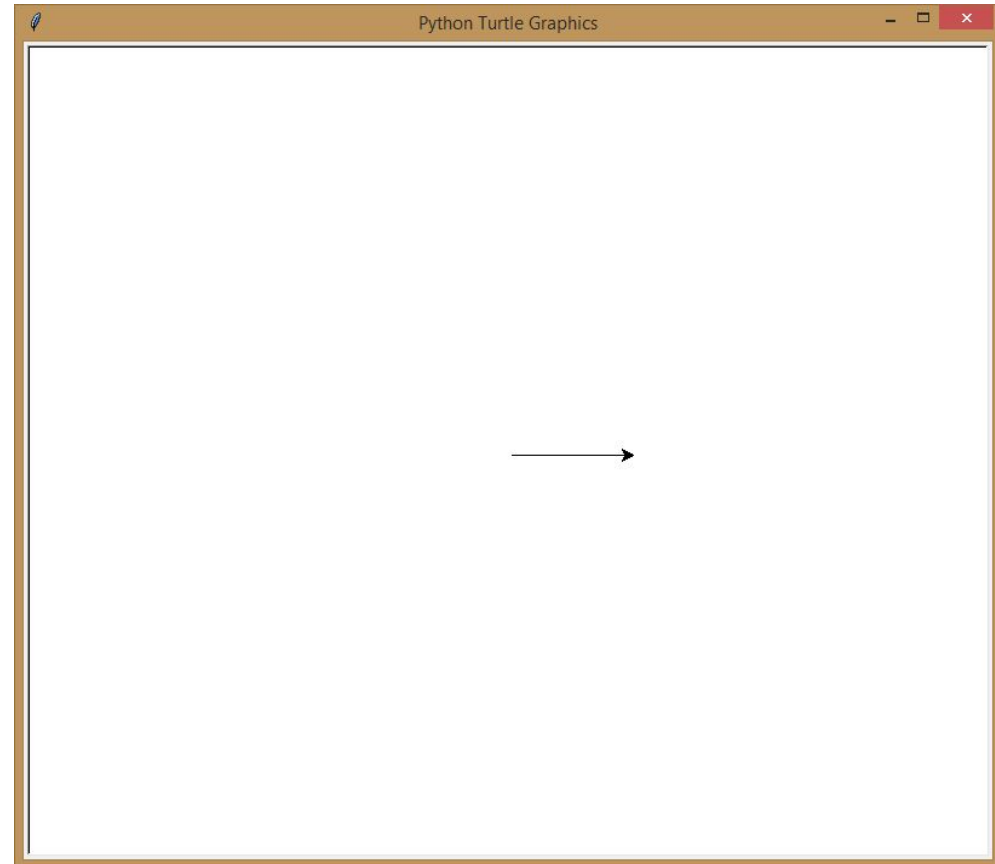
Material Adaptado de Susan Ibach e Christopher Harrison

Na programação, usamos tarefas repetitivas

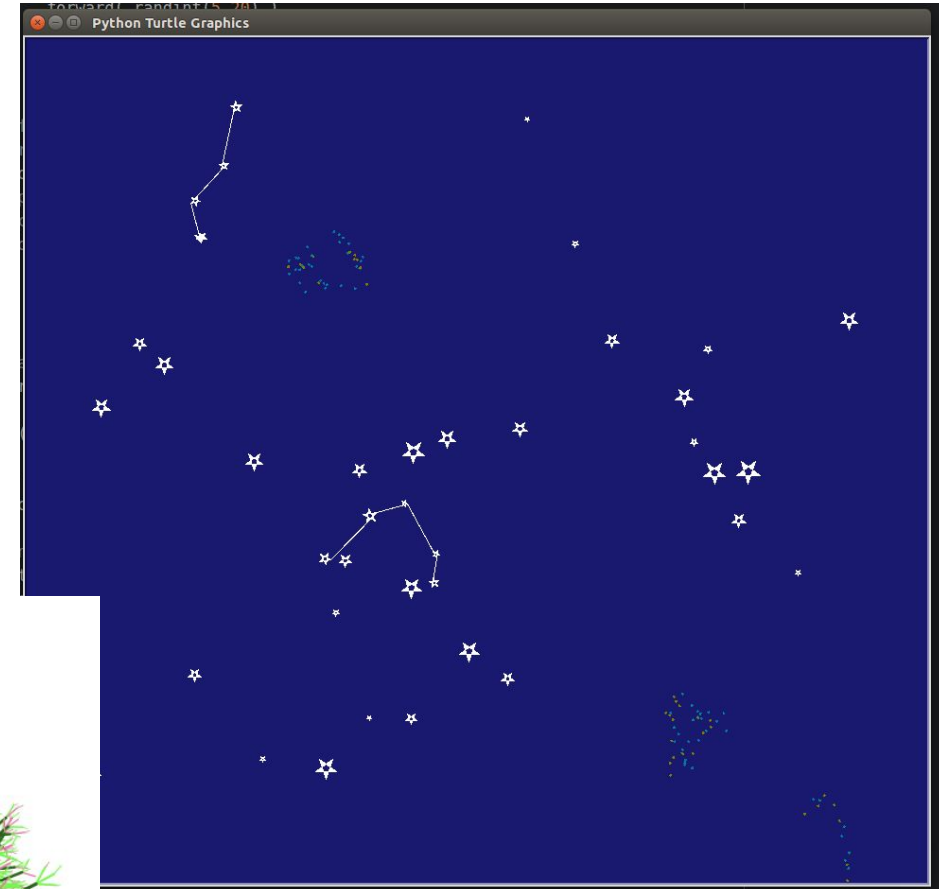
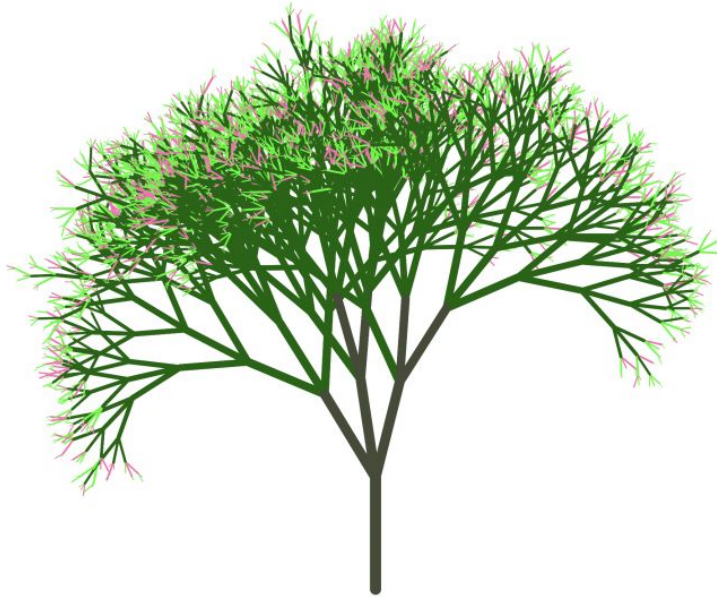
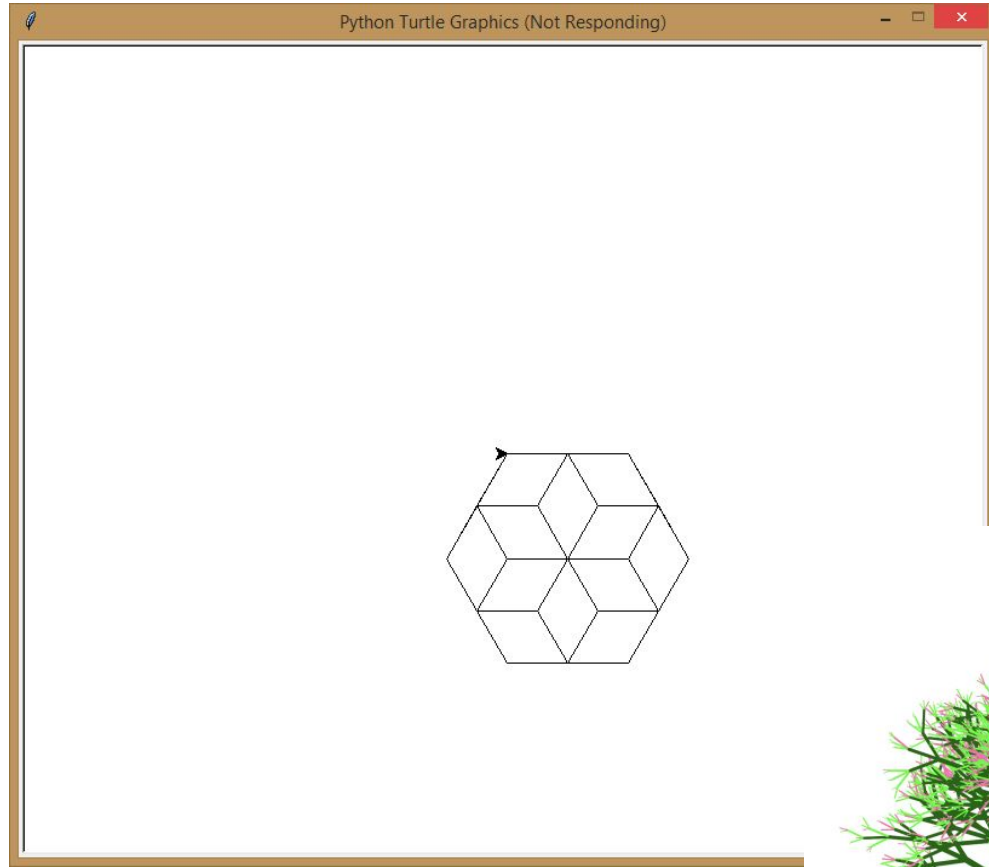
- Contar a média de notas dos alunos da turma
- Verificar quais clientes tem saldo na conta
- Adicionar o juro a a conta corrente dos devedores
- ...

Desenhando com Python

```
from turtle import *  
forward(100)  
done()
```

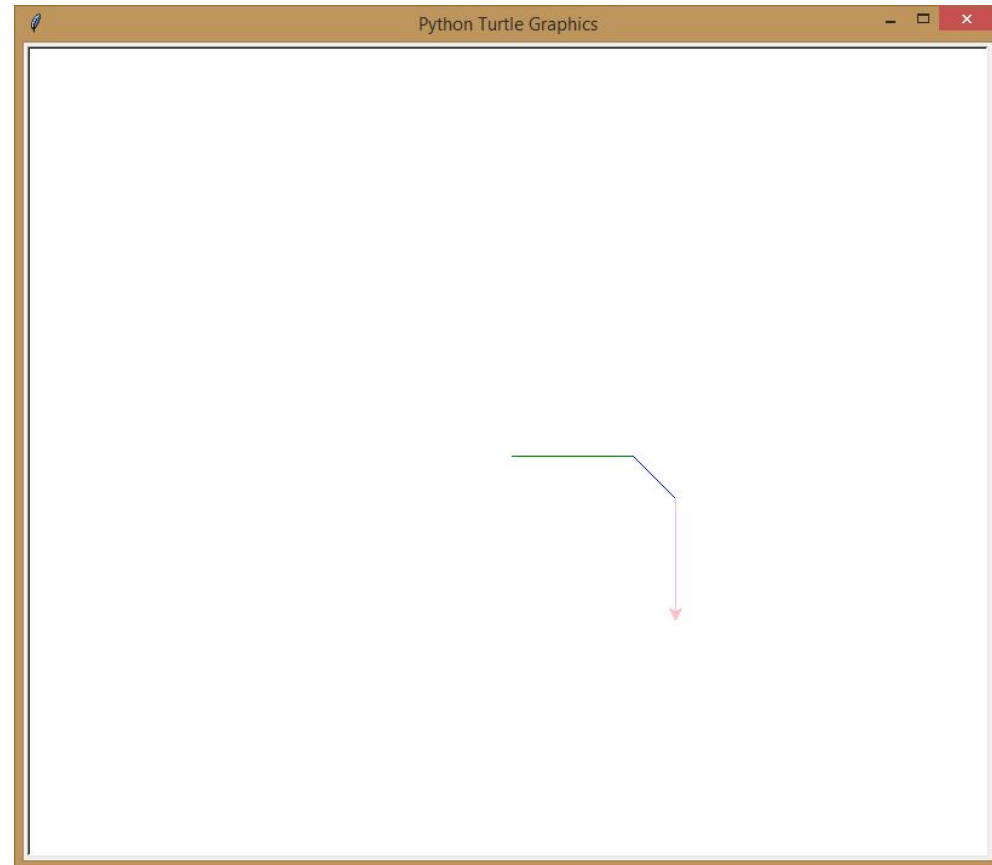


Desenhando com Python



Mais um exemplo

```
from turtle import *  
color('green')  
forward(100)  
right(45)  
color('blue')  
forward(50)  
right(45)  
color('green')  
forward(100)  
done()
```



Instalar o turtle

```
sudo apt-get install python3-pip python3-tk  
pip3 install turtleplus
```

Turtle comandos

Command	Action
<code>right(x)</code>	girar X graus para direita
<code>left(x)</code>	girar X graus para esquerda
<code>color('x')</code>	Muda a cor da caneta
<code>forward(x)</code>	move distância X para frente
<code>backward(x)</code>	move distância X para trás
<code>setpos(x,y)</code>	move para posição x,y
<code>penup/pendown</code>	movimenta a caneta para cima/baixo

<https://docs.python.org/3/library/turtle.html>

Desenhando

```
from turtle import *
```

```
forward(100)
```

```
right(90)
```

```
forward(100)
```

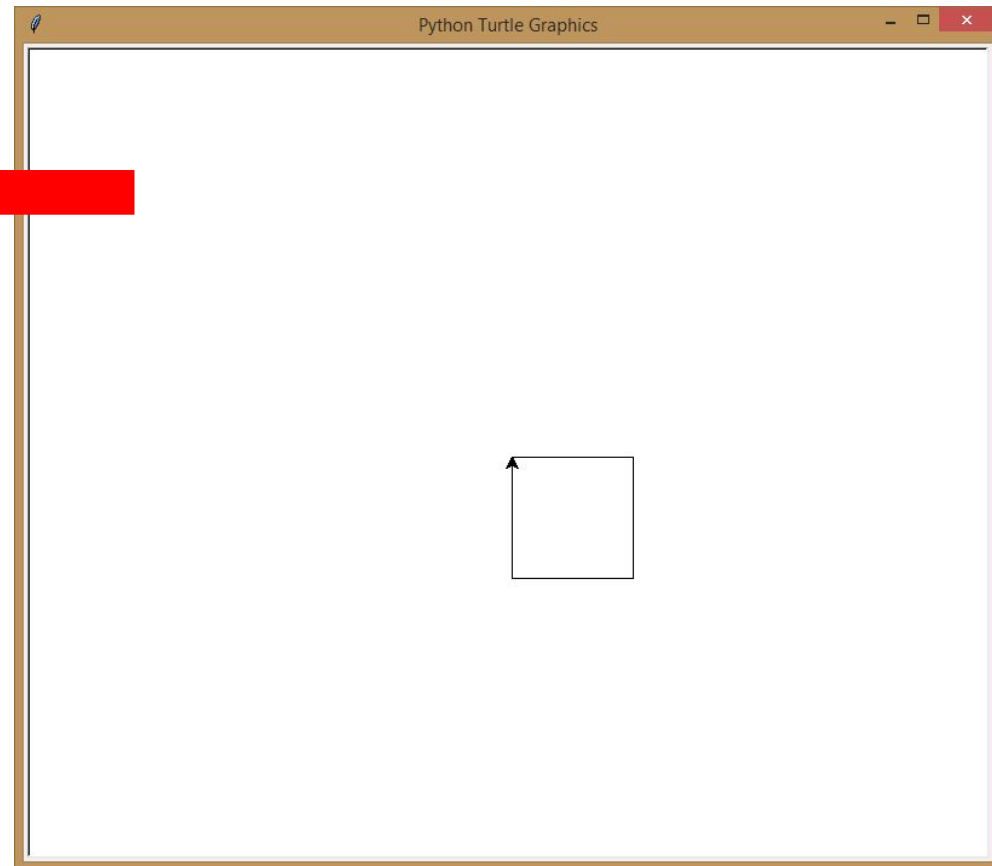
```
right(90)
```

```
forward(100)
```

```
right(90)
```

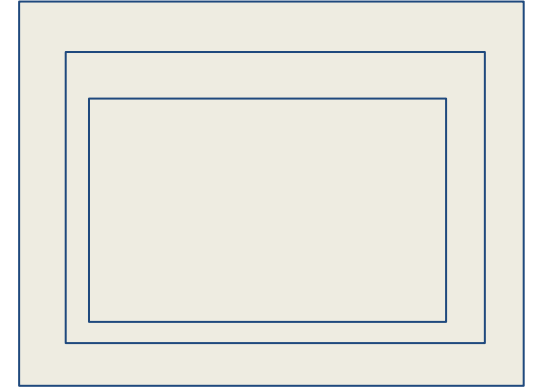
```
forward(100)
```

```
done()
```



Atividade

- 1 - desenhe um quadrado
- 2 - desenhe um triângulo retângulo (100,100,141)
- 3 - desenhe 3 triângulos um dentro do outro
- 5 - desenhe 3 quadrados um dentro do outro.



```
from turtle import *
```

```
forward(100)
```

```
right(90)
```

```
forward(100)
```

```
right(90)
```

```
forward(100)
```

```
right(90)
```

```
forward(100)
```

```
right(90)
```

```
forward(100)
```

```
penup()
```

```
setpos(10,-10)
```

```
pendown()
```

```
forward(80)
```

```
right(90)
```

```
forward(80)
```

```
right(90)
```

```
forward(80)
```

```
right(90)
```

```
forward(80)
```

```
right(90)
```

```
forward(80)
```

```
done()
```

```
from turtle import *
```

```
i=0
```

```
left(90)
```

```
forward(100)
```

```
right(135)
```

```
forward(141)
```

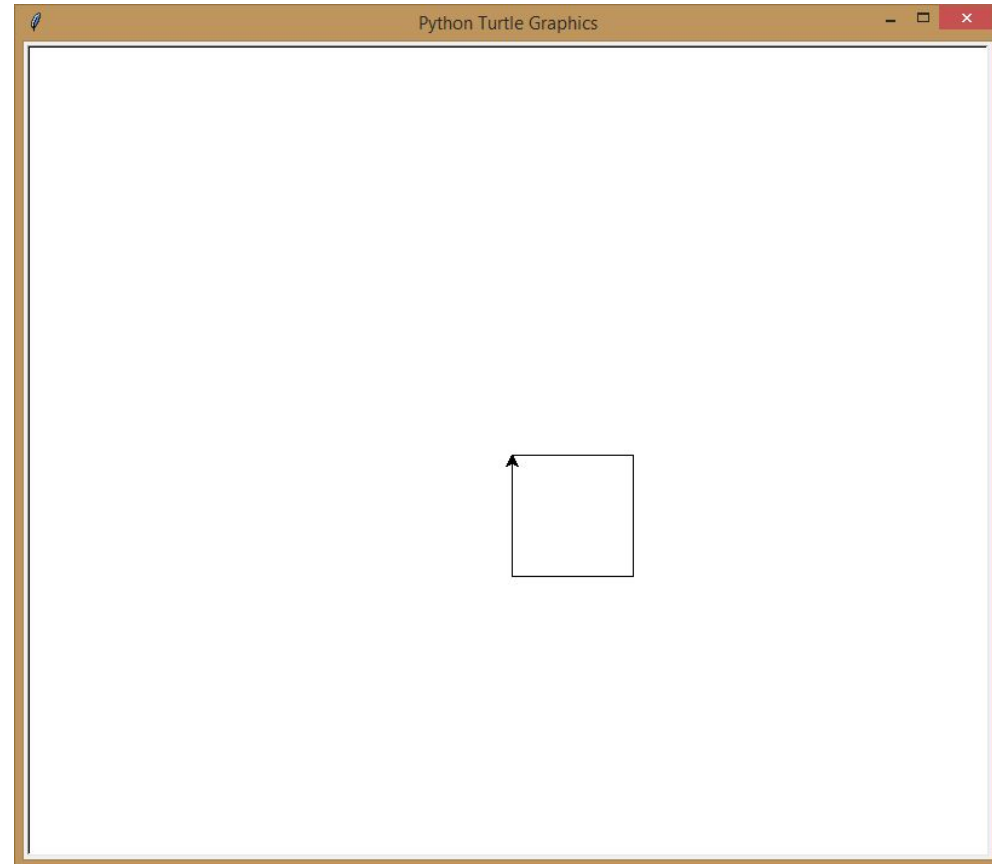
```
right(135)
```

```
forward(100)
```

```
done()
```

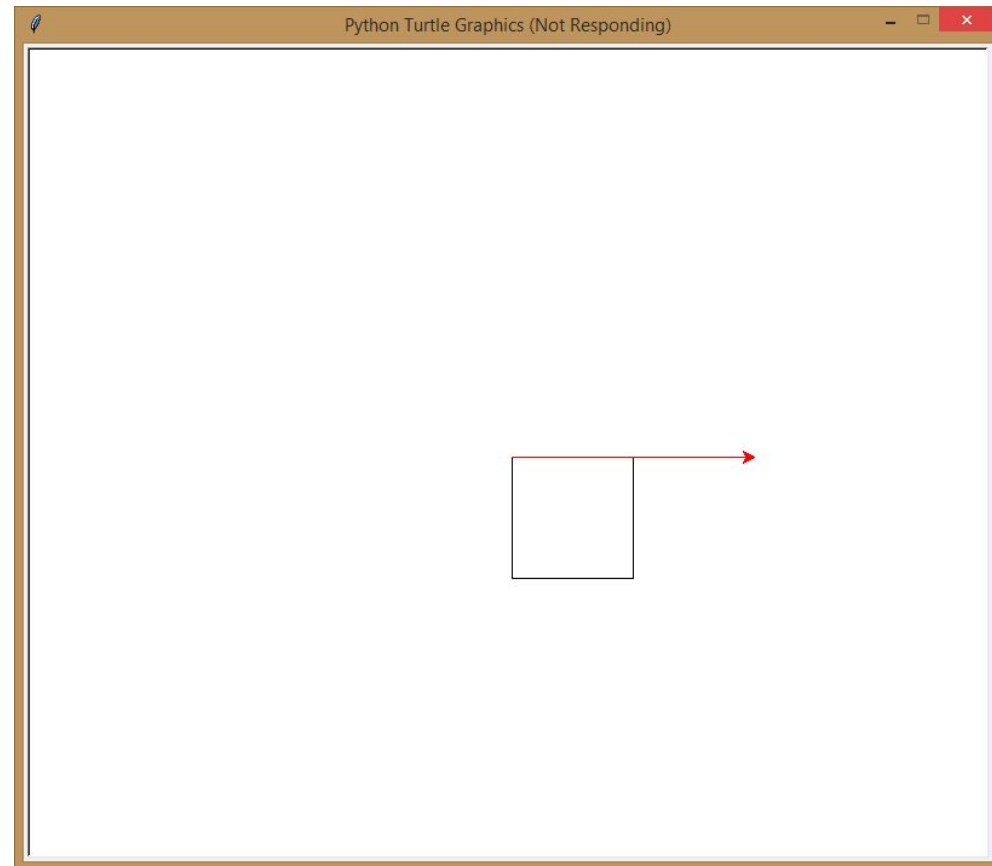
Comandos de repetição

```
from turtle import *  
i=0  
while (i<4):  
    forward(100)  
    right(90)  
    i=i+1
```



SOMente código indentado será executado no loop

```
from turtle import *
i=0
while i<4:
    forward(100)
    right(90)
    i=i+1
color('red')
forward(200)
```



Podemos cometer outros erros

Quais são os erros:

```
from turtle import *  
i=0  
while i<4:  
    forward(100)  
    right(90)  
    i+=1  
done()
```

Atividade

- 1 - Agora desenhe 10 quadrados um dentro do outro (usando um loop de repetição)
- 2 Construa 10 triângulos de diferentes tamanhos usando loop
- 3 Use sua criatividade!

```
from turtle import *  
i=0
```

```
while i<4:  
    forward(100)  
    right(90)  
    i=i+1
```

```
setpos(10,-10)
```

```
i=0
```

```
while i<4:  
    forward(80)  
    right(90)  
    i=i+1
```

```
done()
```


Agora desenhe 10 quadrados um dentro do outro usando um loop de repetição

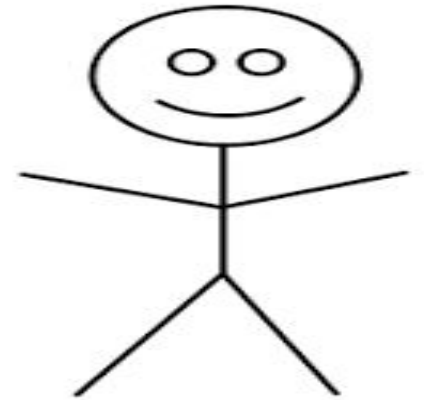
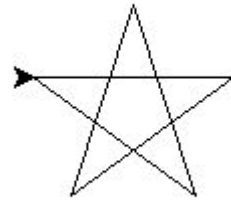
```
from turtle import *  
for steps in range(4):  
    forward(100)  
    right(90)  
goto(10,-10)  
for steps in range(4):  
    forward(80)  
    right(90)
```

Avançando um pouco no Turtle

```
from turtle import *  
canvas = Screen()  
canvas.setup(400,200)  
pedro = Turtle()  
pedro.left(180)  
pedro.forward(50)  
canvas.exitonclick()
```

Atividade

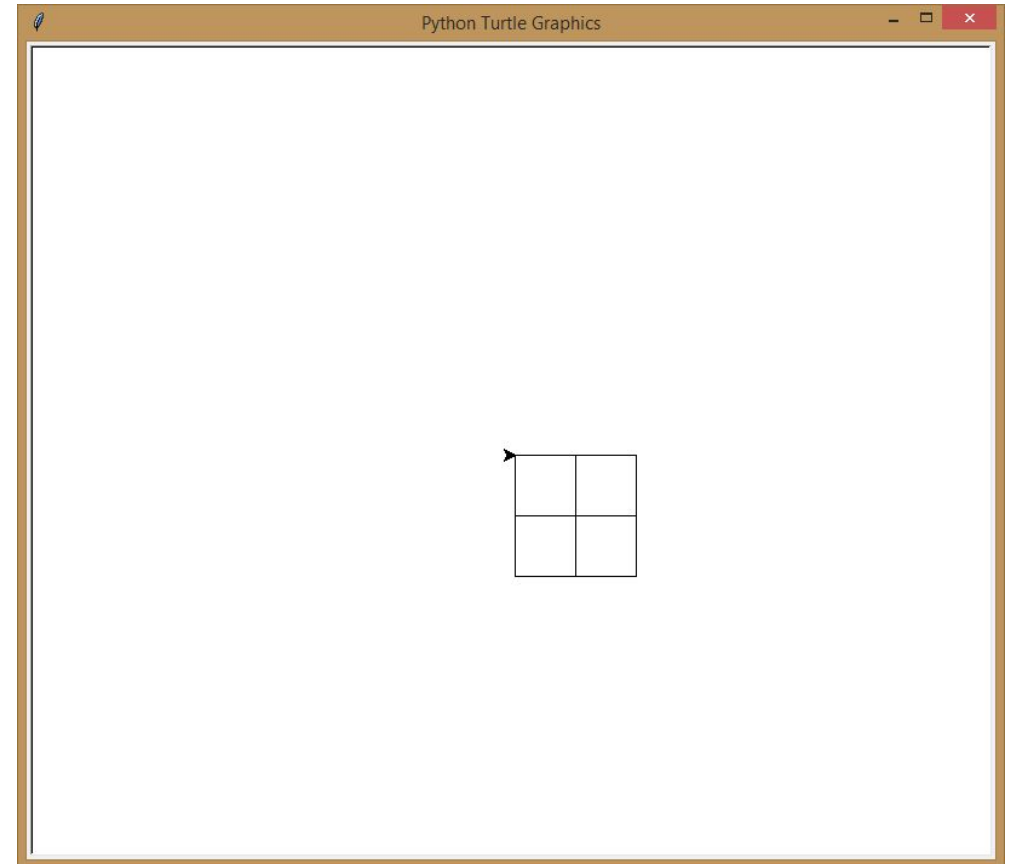
- 1- Usando o canvas desenhe um círculo
- 2- Usando o canvas desenhe um círculo dentro de outro círculos
- 3- Desenhe figura a seguir



Desafio: desenhe um boneco conforme a figura

Loops aninhados

```
from turtle import *
i=0
while i<4:
    forward(100)
    right(90)
    i+=1
    j=0
    while j<4:
        forward(50)
        right(90)
        j+=1
```



Usando variáveis para definir o tamanho

```
import turtle
n_lados = 6
i=0
while (i<n_lados):
    turtle.forward(100)
    turtle.right(360/n_lados)
    i+=1

turtle.done()
```

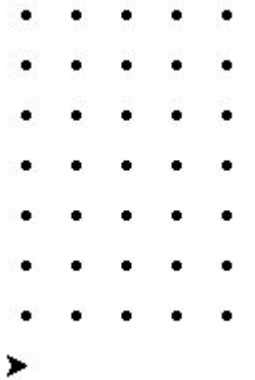
Loops

1 - desenhe um círculo

2 - Faça a seguinte figura usando loops (comando dot(10) cria um ponto)



2 - Faça a seguinte figura usando loops aninhados ->



3 - Faça a seguinte figura usando loops aninhados

->



Loops

```
for steps in range(4) :  
    print(steps)
```

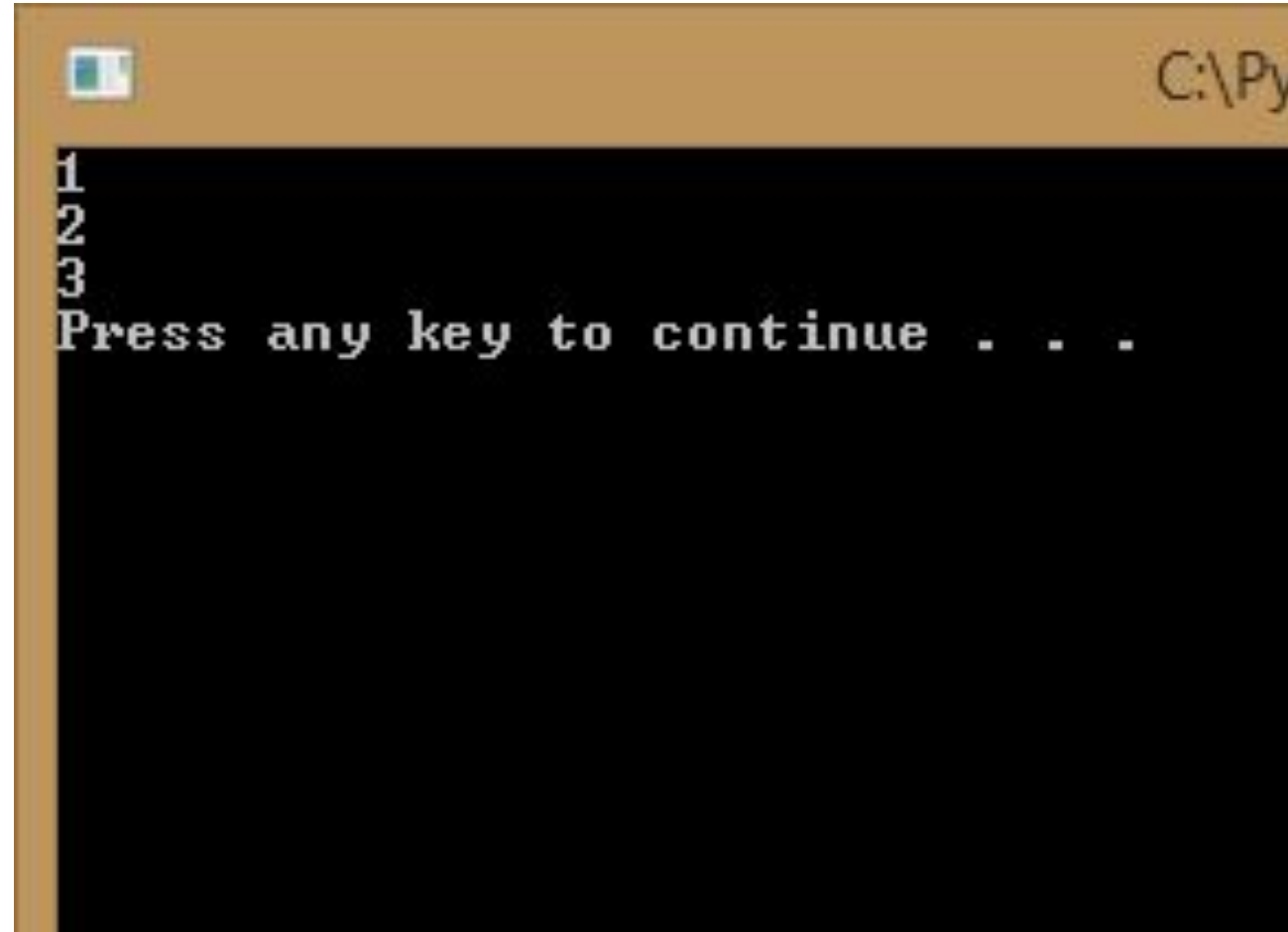


A screenshot of a Python terminal window. The title bar at the top right shows the file path "C:\Python34\". The terminal output displays the numbers 0, 1, 2, and 3 on separate lines, followed by the prompt "Press any key to continue . . .".

```
C:\Python34\  
0  
1  
2  
3  
Press any key to continue . . .
```

Loops

```
for steps in range(1,4) :  
    print(steps)
```

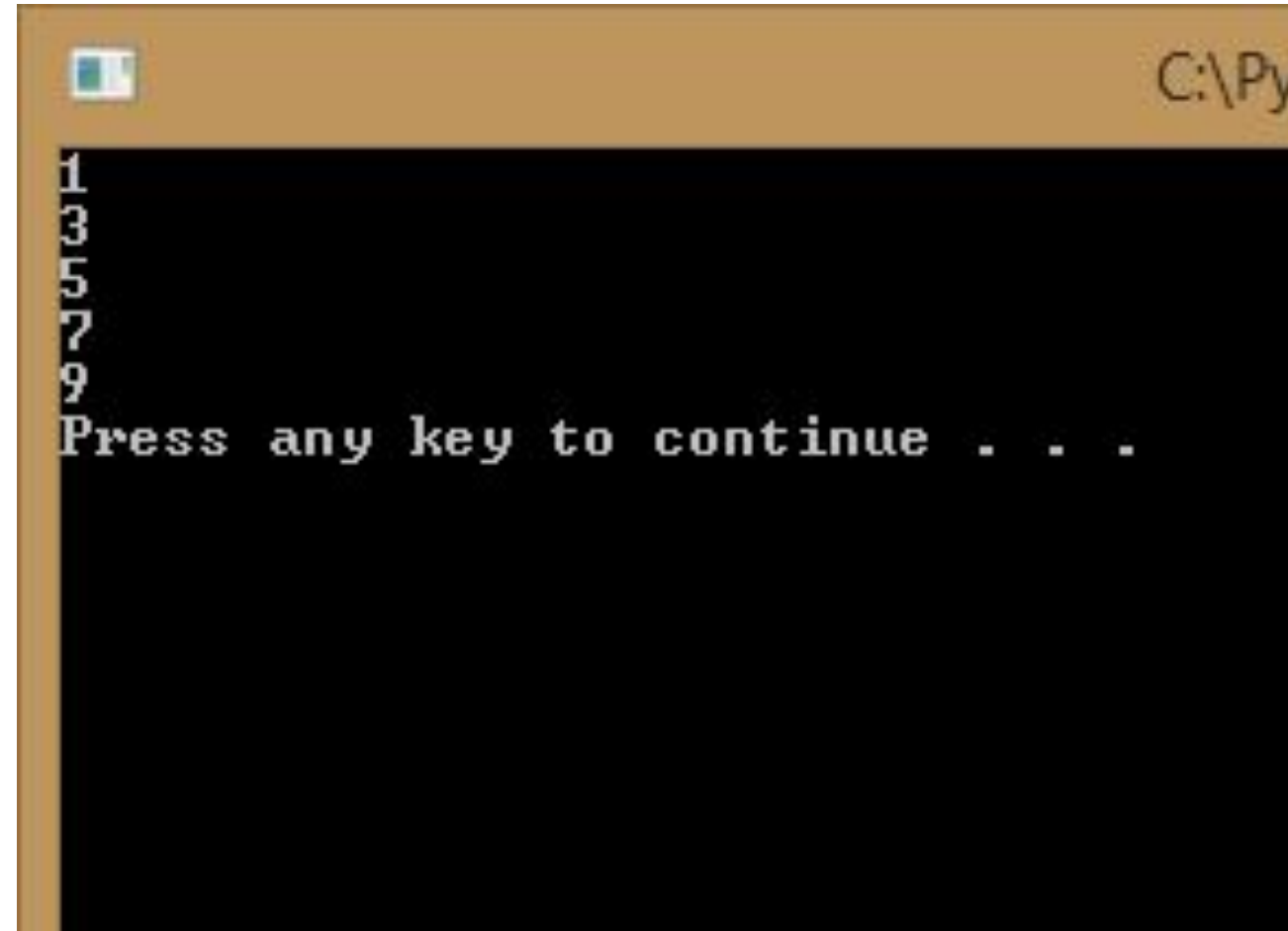


A screenshot of a Windows command prompt window. The title bar is brown and shows a small icon on the left and the path 'C:\P...' on the right. The command prompt has a black background with white text. It displays the numbers '1', '2', and '3' on three separate lines, which are the outputs of the Python code shown on the left. Below these numbers, the text 'Press any key to continue . . .' is displayed.

```
1  
2  
3  
Press any key to continue . . .
```


Loops

```
for steps in range(1,10,2) :  
    print(steps)
```




A screenshot of a Python interpreter window. The window has a title bar with a small icon on the left and the path 'C:\P...' on the right. The main area is black with white text. The output of the loop is shown as a vertical list of numbers: 1, 3, 5, 7, 9. Below the numbers, the text 'Press any key to continue . . .' is displayed.

```
C:\P...  
1  
3  
5  
7  
9  
Press any key to continue . . .
```

Loops

```
for steps in [1,2,3,4,5]:  
    print(steps)
```



A screenshot of a Windows command prompt window. The title bar is brown and shows a small icon on the left and the path 'C:\Pj' on the right. The command prompt has a black background with white text. It displays the numbers 1, 2, 3, 4, and 5 on separate lines, which are the outputs of the Python code shown in the previous block. Below the numbers, it says 'Press any key to continue . . .'.

```
1  
2  
3  
4  
5  
Press any key to continue . . .
```

loops

```
import turtle
for steps in ['red', 'blue', 'green', 'black'] :
    turtle.color(steps)
    turtle.forward(100)
    turtle.right(90)
```

