

Introdução

Prática

Jan K. S. – janks@puc-rio.br

ENG 1419 – Programação de Microcontroladores

Forward 20 steps! Now turn
right, by 45 degrees! Now
go back 40 steps! Turn
right, 90 degrees!



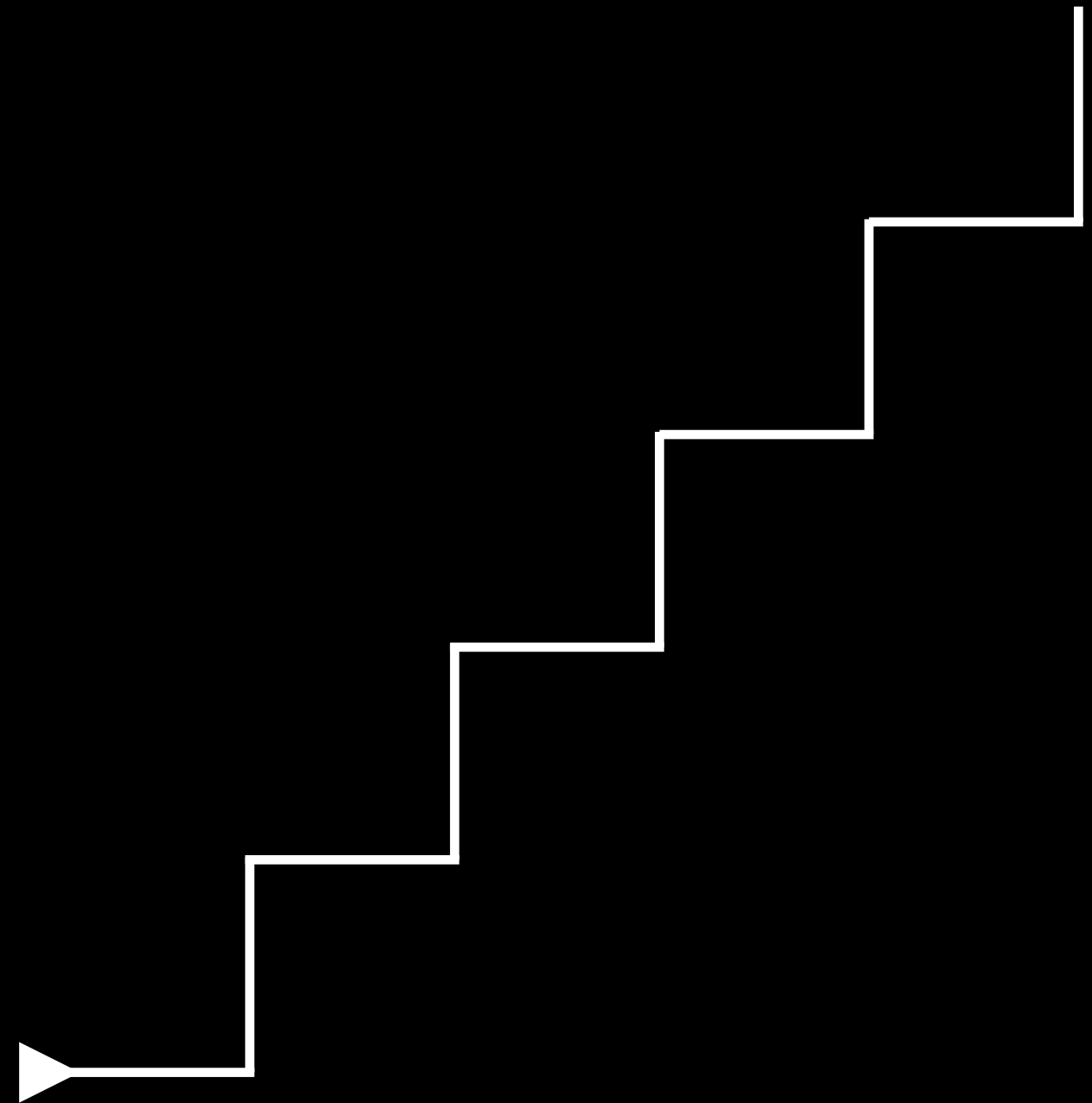
revised To: The Sea.com

```
>>> from turtle import *
>>> forward(100)    # avance 100 pontos de distância
>>> right(90)       # vire 90° à direita
>>> forward(100)
>>> right(90)
>>> forward(100)
>>> right(90)
>>> forward(100)
```



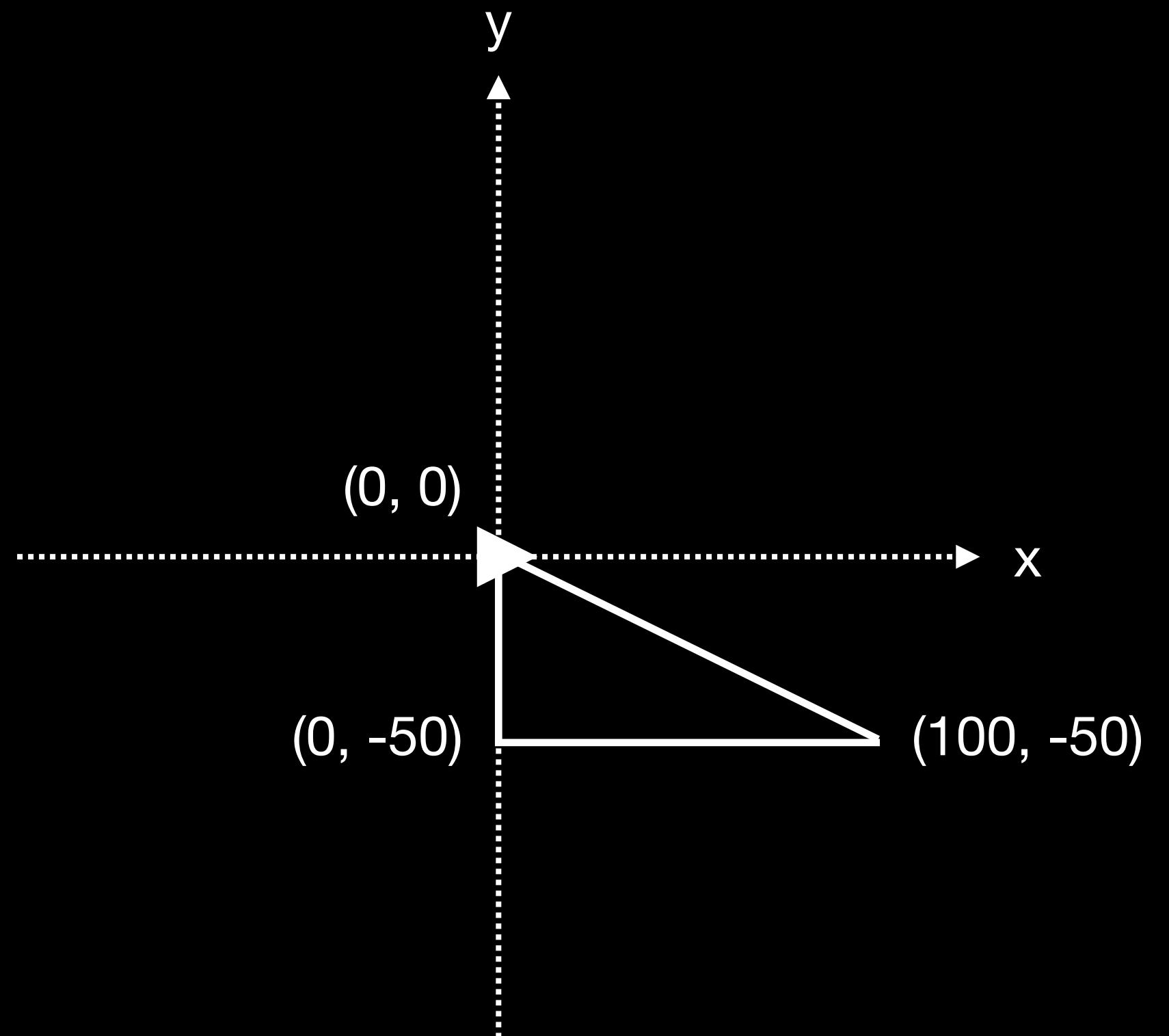
Comandos para Avançar e Rotacionar

```
>>> from turtle import *
>>> for i in range(0, 7):
...     forward(50)
...     left(90)
...     forward(50)
...     right(90)
... 
```



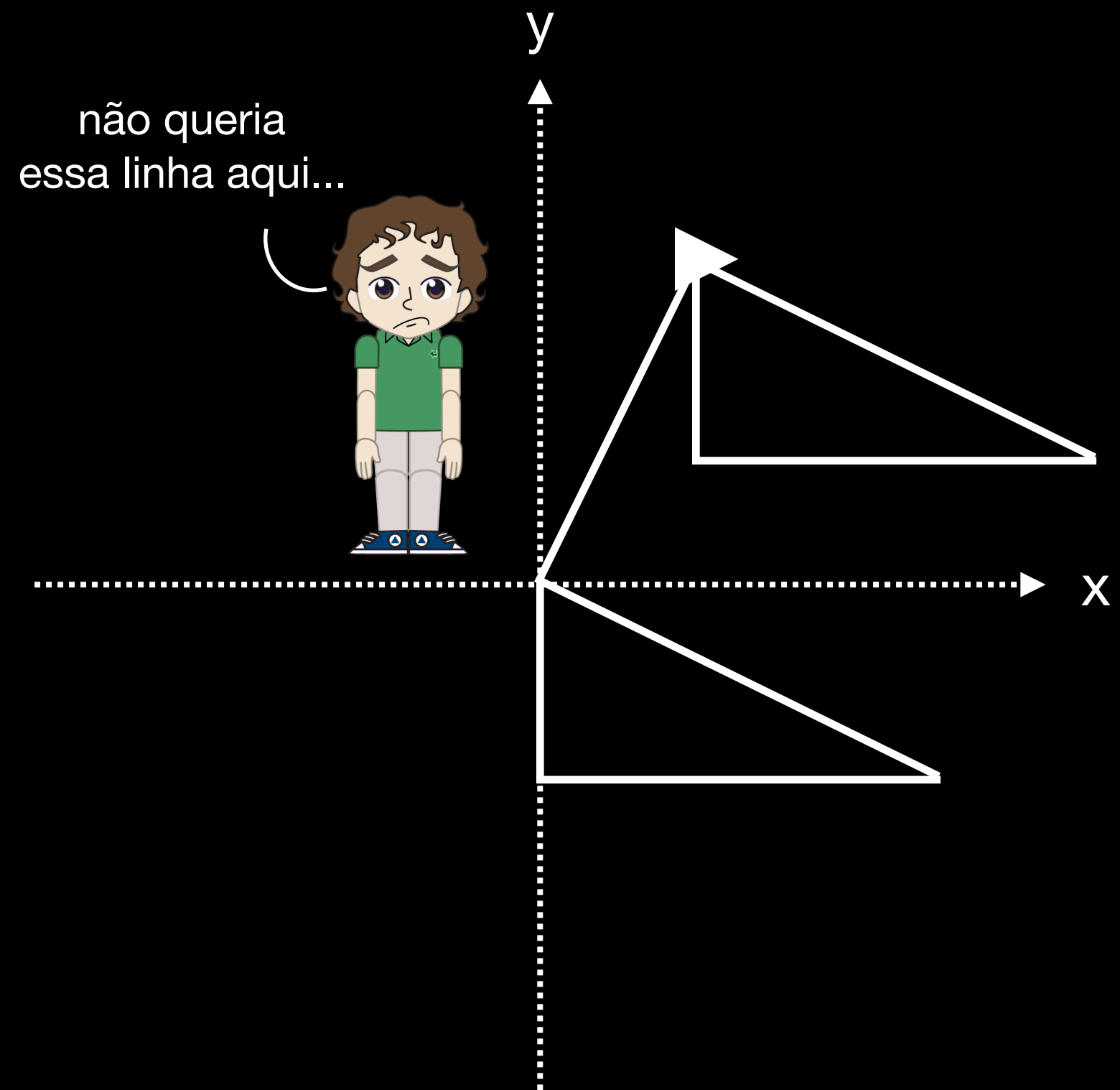
Rotação para Esquerda e para a Direita

```
>>> from turtle import *  
>>> goto(0, -50)  
>>> goto(100, -50)  
>>> goto(0, 0)
```



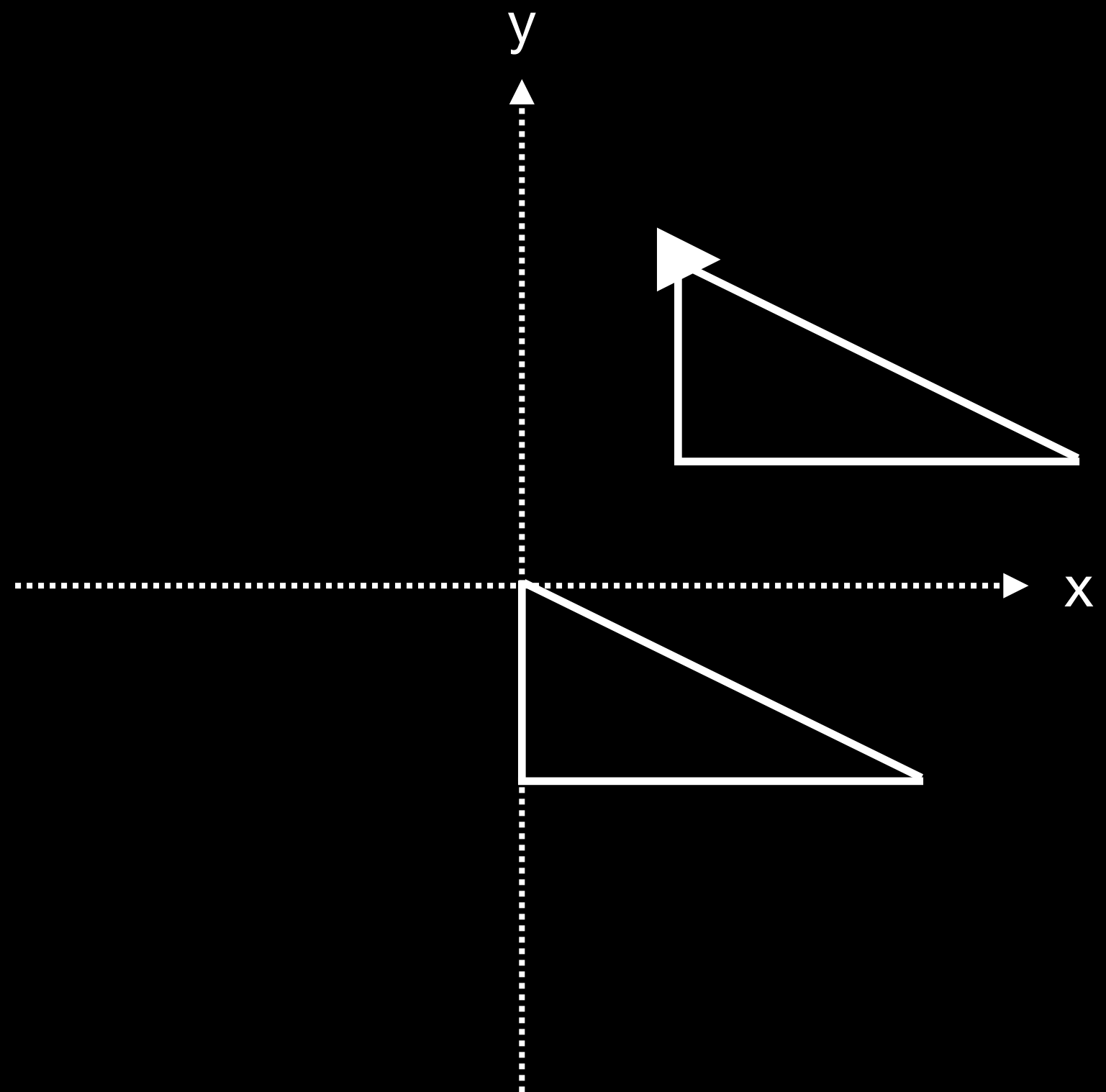
Deslocamento para um Ponto Específico

```
>>> from turtle import *
>>> goto(0, -50)
>>> goto(100, -50)
>>> goto(0, 0)
>>> goto(40, 80)
>>> goto(40, 30)
>>> goto(140, 30)
>>> goto(40, 80)
```



Desenho de uma Segunda Figura

```
>>> from turtle import *
>>> goto(0, -50)
>>> goto(100, -50)
>>> goto(0, 0)
>>> penup()
>>> goto(40, 80)
>>> pendown()
>>> goto(40, 30)
>>> goto(140, 30)
>>> goto(40, 80)
```



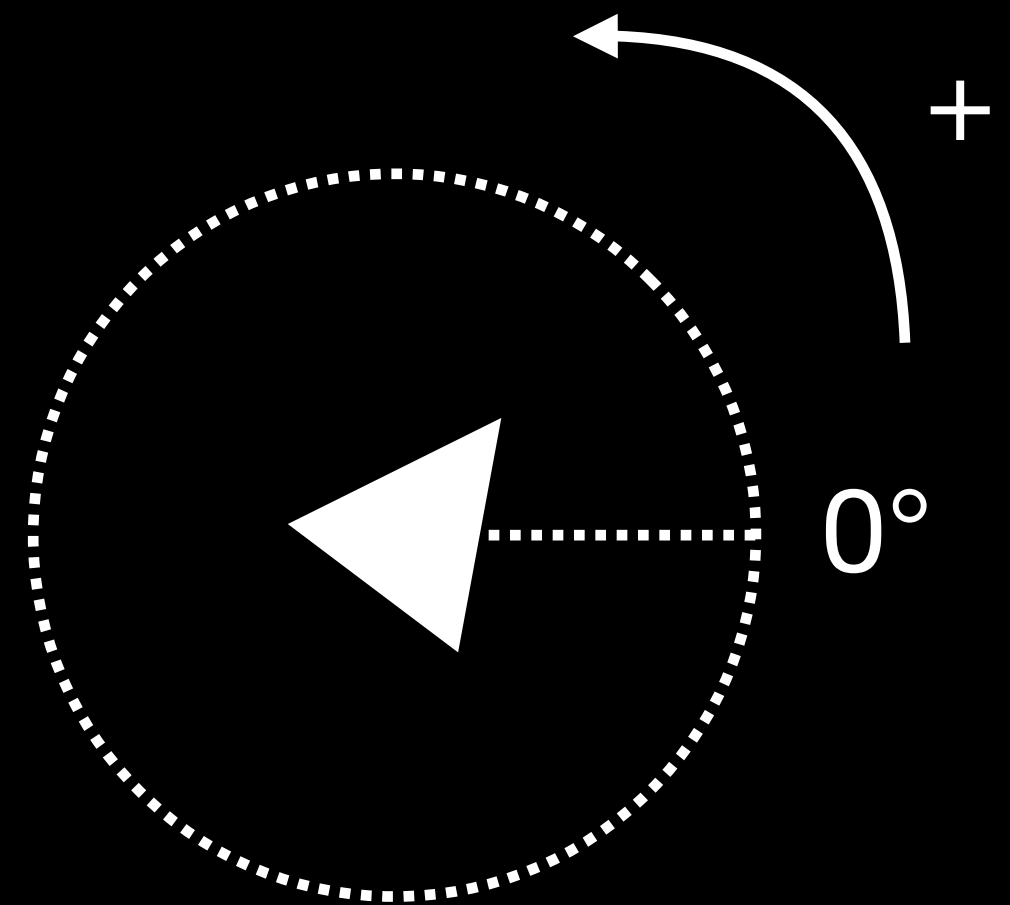
Desenho de uma Segunda Figura "Levantando" e "Abaixando" a Caneta

```
>>> from turtle import *
```

```
>>> ...
```

```
>>> setheading(0)
```

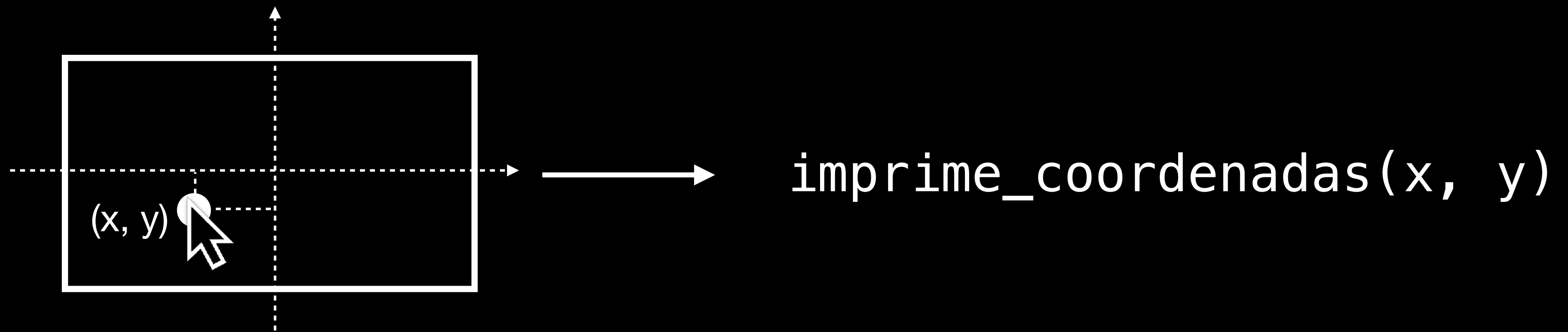
define ângulo como 0° ,
independentemente do ângulo inicial



Definição Absoluta do Ângulo

▶ Olá, mundo!

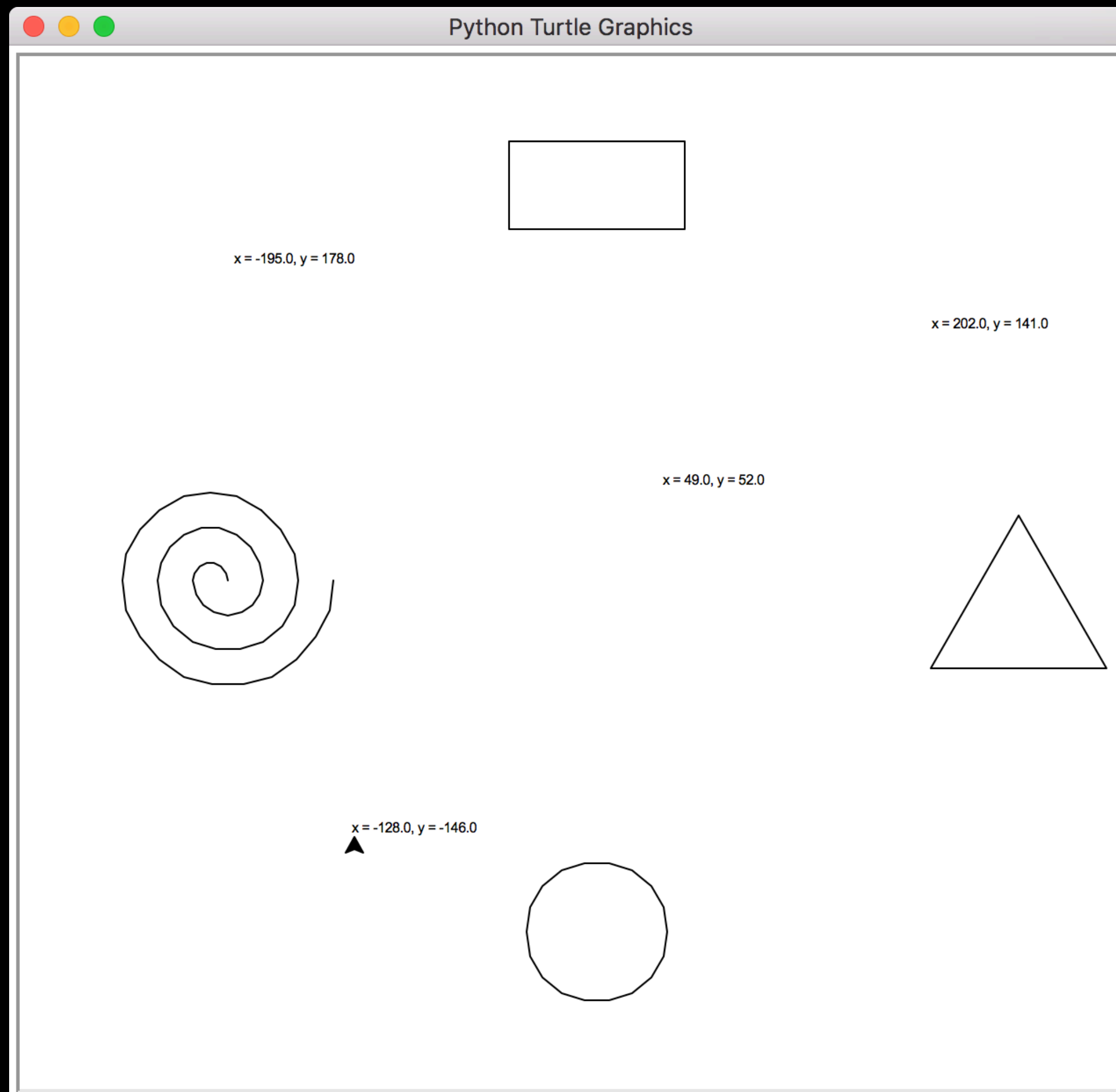
```
>>> write("Olá, mundo!")
```



```
>>> def imprime_coordendas(x, y):  
...     print("x = ", x)  
...     print("y = ", y)  
...  
>>> onscreenclick(imprime_coordendas)
```

não coloque parênteses no final!

Testes Iniciais



Objetivo dos Testes Iniciais



Testes Iniciais

Desenhe o retângulo 100x50 no topo.

↳ DICA: adapte o exemplo do quadrado nos slides.

Desenhe um triângulo equilátero na direita.

↳ DICA: use a função `setheading` para definir o ângulo.

Desenhe o círculo na parte de baixo.

↳ DICA: consulte o [site da documentação oficial](#).

Desenhe a espiral na esquerda.

↳ DICA: desenhe semi-círculos aumentando o raio.

Ao clicar em um ponto da tela, **desenhe um texto** com o valor das coordenadas x e y , no formato " $x = -32.0, y = 231.0$ ".

↳ DICA: use a função `write` e a concatenação de string com números.

Implementação



Desenho de Bandeiras

```
>>> from turtle import *
```

```
>>> forward(100)
```

```
>>> right(90)
```

```
>>> forward(100)
```

```
>>> right(90)
```

```
>>> forward(100)
```

```
>>> right(90)
```

```
>>> forward(100)
```



Exemplo Anterior do Quadrado


```
>>> from turtle import *
>>> fillcolor("blue")    # define cor de preenchimento
>>> begin_fill()         # inicia preenchimento
>>> forward(100)
>>> right(90)
>>> forward(100)
>>> right(90)
>>> forward(100)
>>> right(90)
>>> forward(100)
>>> end_fill()          # finaliza preenchimento
```



Exemplo Anterior do Quadrado com Preenchimento

Thonny - /Users/Jan/Library/Mobile Documents/com~apple~CloudDocs/PUC/Microncontroladores/Introdução/Introducao/implementacao.py @ 31 : 82

implementacao.py

```
from turtle import *

def desenha_retangulo(x, y, comprimento, altura, cor):
    return

def desenha_circulo(x, y, raio, cor):
    return

def desenha_poligono(lista_pontos, cor):
    return

# Bandeira 1
desenha_retangulo(0, 40, 100, 20, 'blue')
desenha_retangulo(0, 20, 100, 20, 'white')
desenha_retangulo(0, 0, 100, 20, 'red')

# Bandeira 2
desenha_retangulo(0, 140, 100, 20, 'orange')
desenha_retangulo(0, 120, 100, 20, 'white')
desenha_retangulo(0, 100, 100, 20, 'green')
desenha_circulo(50, 110, 10, 'orange')

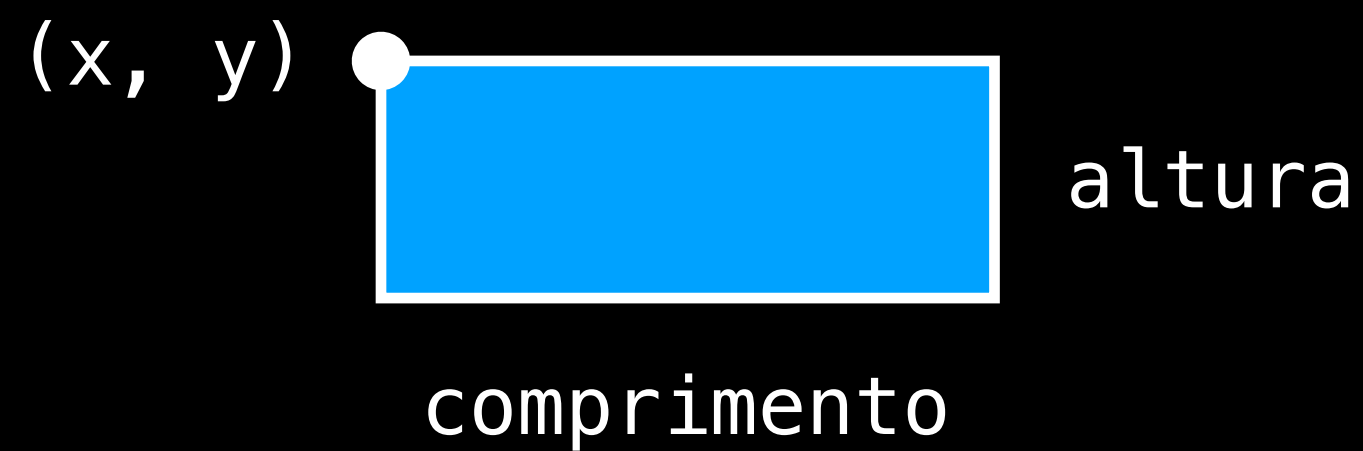
# Bandeira 3
desenha_retangulo(0, 230, 100, 30, 'white')
```

Shell

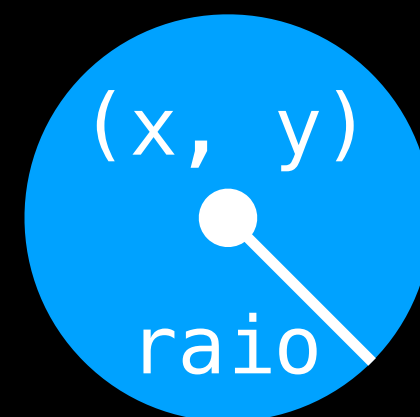
```
>>>
```

Objetivo: Implementar Funções Auxiliares de Desenho das Bandeiras

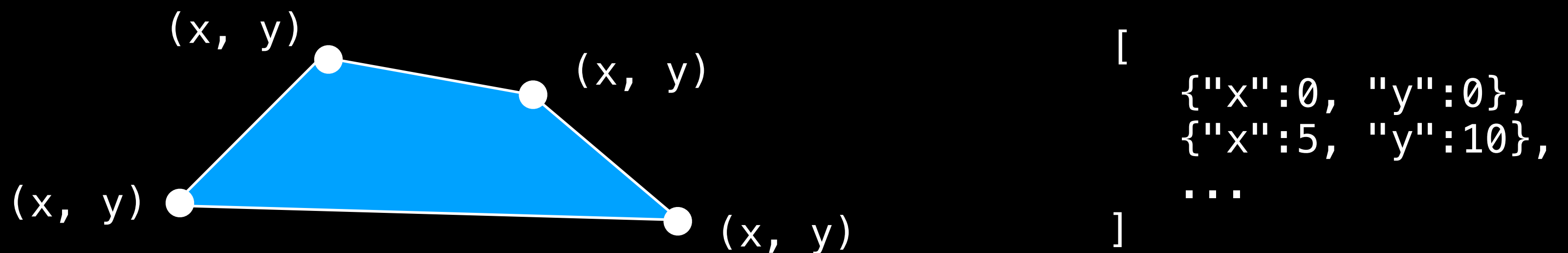
`desenhar_retangulo(x, y, comprimento, altura, cor)`



`desenhar_circulo(x, y, raio, cor)`



`desenhar_poligono(lista_de_pontos, cor)`



Descrição das Funções



Implementação

Implemente a função `desenha_retangulo`.

↳ DICA: reaproveite o código dos testes iniciais.

Implemente a função `desenha_circulo`.

↳ DICA: tome cuidado com a posição do centro.

Implemente a função `desenha_poligono`.

↳ DICA: vá para o primeiro ponto, passe pelos demais e depois volte ao primeiro.

Aperfeiçoamento

```

{
  "nome": "Brasil",
  "elementos": [
    {
      "tipo": "retângulo",
      "x": -50,
      "y": 30,
      "comprimento": 100,
      "altura": 60,
      "cor": "green"
    },
    {
      "tipo": "polígono",
      "pontos": [
        {"x": 0, "y": 25},
        {"x": -45, "y": 0},
        {"x": 0, "y": -25},
        {"x": 45, "y": 0}
      ],
      "cor": "yellow"
    },
    {
      "tipo": "círculo",
      "x": 0,
      "y": 0,
      "raio": 10,
      "cor": "blue"
    }
  ]
}

```

Dicionário com Dados do País



paises.json



```
[
  {
    "nome": "Brasil",
    "elementos": [
      {
        "tipo": "retângulo",
        ...
      },
      {
        "tipo": "polígono",
        ...
      },
      ...
    ]
  },
  {
    "nome": "Índia",
    "elementos": [
      {
        "tipo": "retângulo",
        ...
      },
      ...
    ]
  },
  ...
]
```

Lista de Dicionário de Países

Thonny - /Users/Jan/Library/Mobile Documents/com~apple~CloudDocs/PUC/Microncontroladores/Introduç...

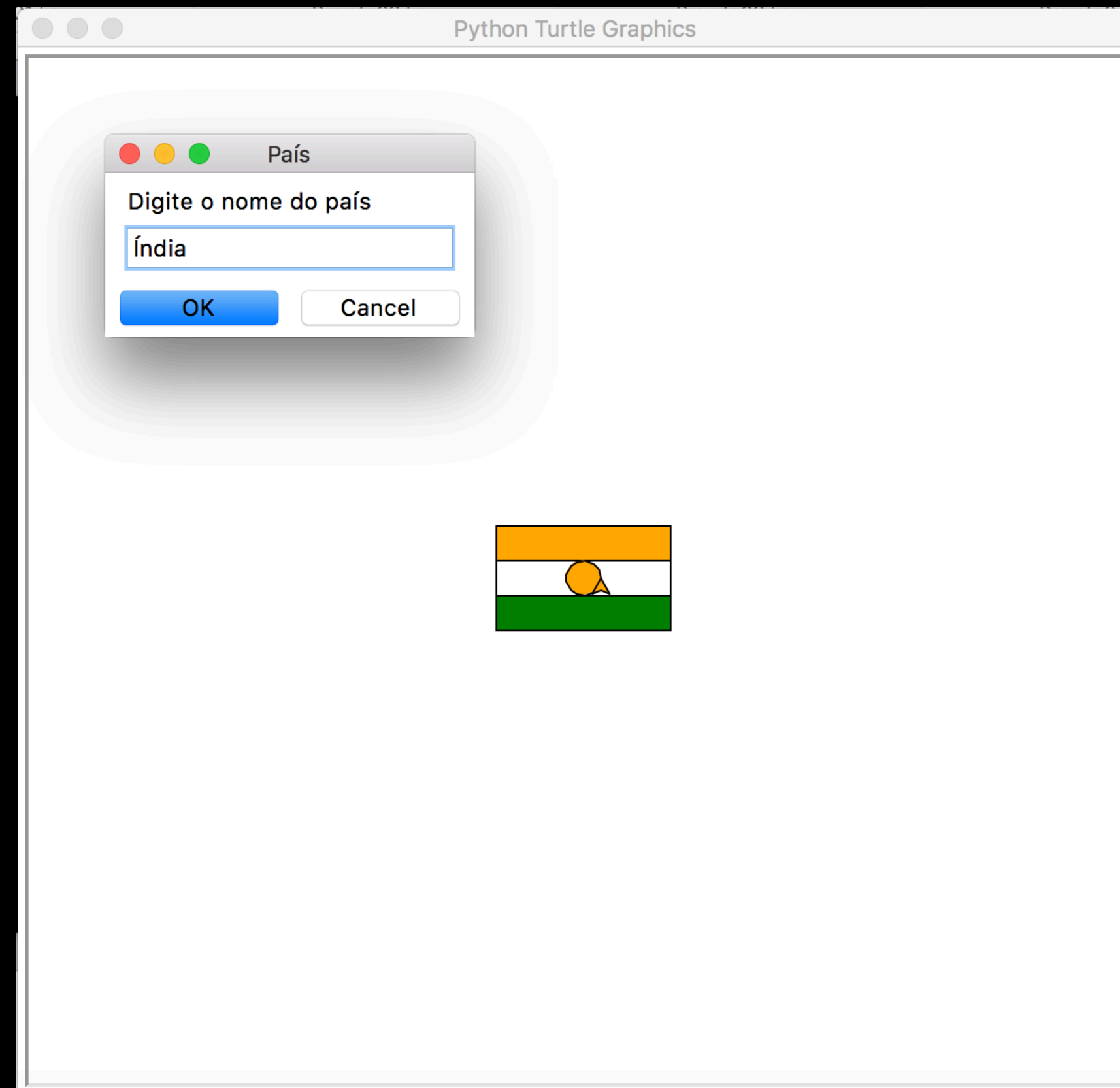
aperfeicoamento.py ×

```
1  from json import load
2  from turtle import *
3
4  # Copie as funções da Implementação aqui
5
6
7  # Implemente a função abaixo
8  def desenha_bandeira(lista_de_elementos):
9      return
10
11  dicionario = load(open('pais_misterioso.json'))
12  desenha_bandeira(dicionario)
13
14  # Crie um arquivo "nova_bandeira.json" com dados de uma nova bandeira e desenhe-a
```

Shell

```
>>>
```

Implementação da Função desenha_bandeira



Escolha do País a Ser Desenhado



Aperfeiçoamento

Implemente a função **desenha_bandeira**.

↳ DICA: acesse o campo "elementos" do dicionário, percorra esses elementos e use as funções da Implementação.

Ao clicar em um ponto da tela, use a função **textinput** para solicitar o nome de um país. Em seguida, busque esse país na lista de dicionário e desenhe-o usando a função anterior.

↳ DICA: percorra a lista de dicionários e compare o campo "nome" com o país solicitado.

Adicione uma nova bandeira no arquivo JSON e teste seu desenho.

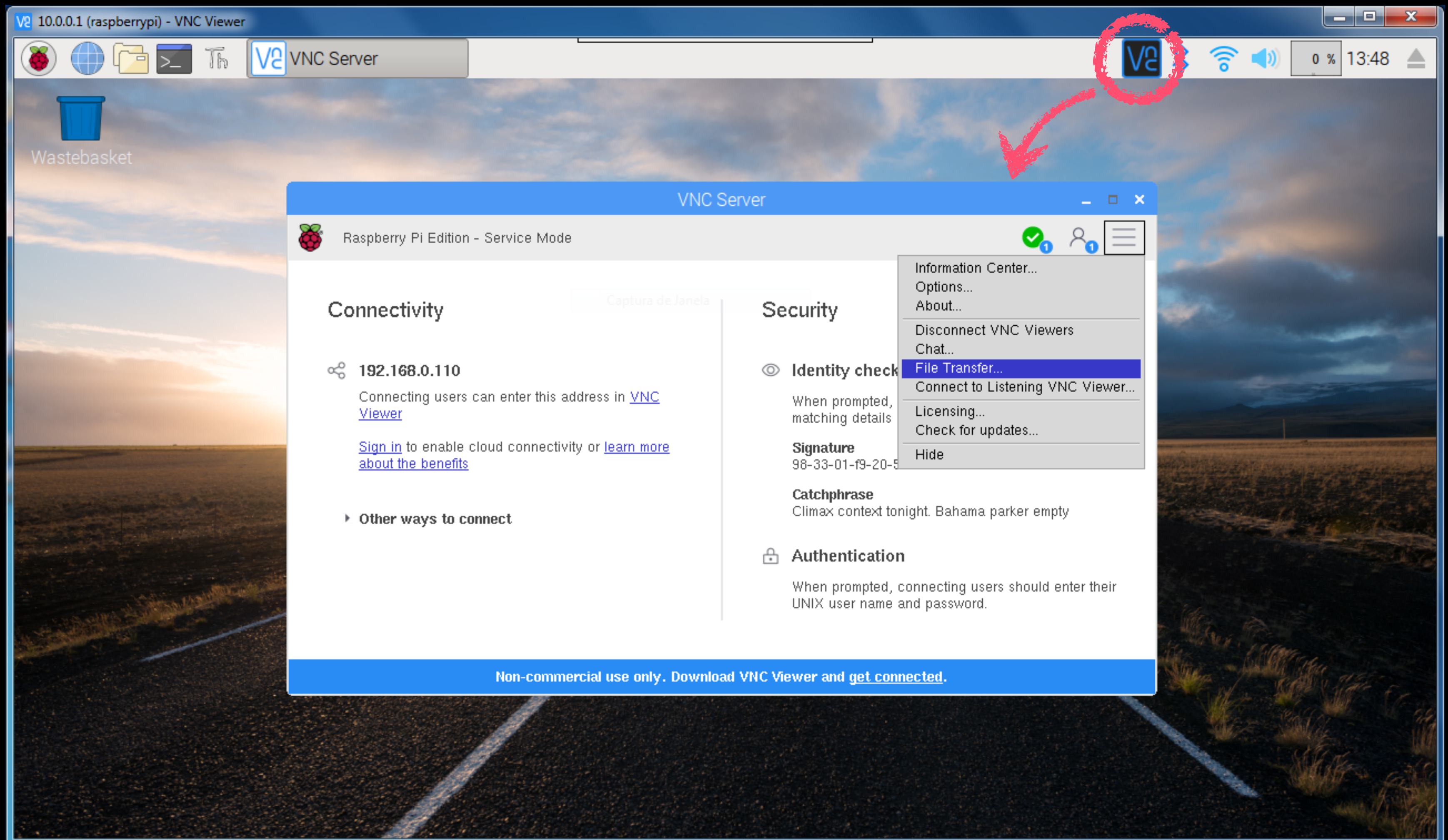


Material da Introdução



janks.link/micro/introducao.zip

Material da Introdução



Transferência de Arquivos entre PC e Raspberry Pi via VNC