

海龟岛的试炼（二）





1. 探索新知

1.1

circle用法进阶

```
turtle.circle(radius, extent=None, steps=None)
```



圆



半径



圆弧

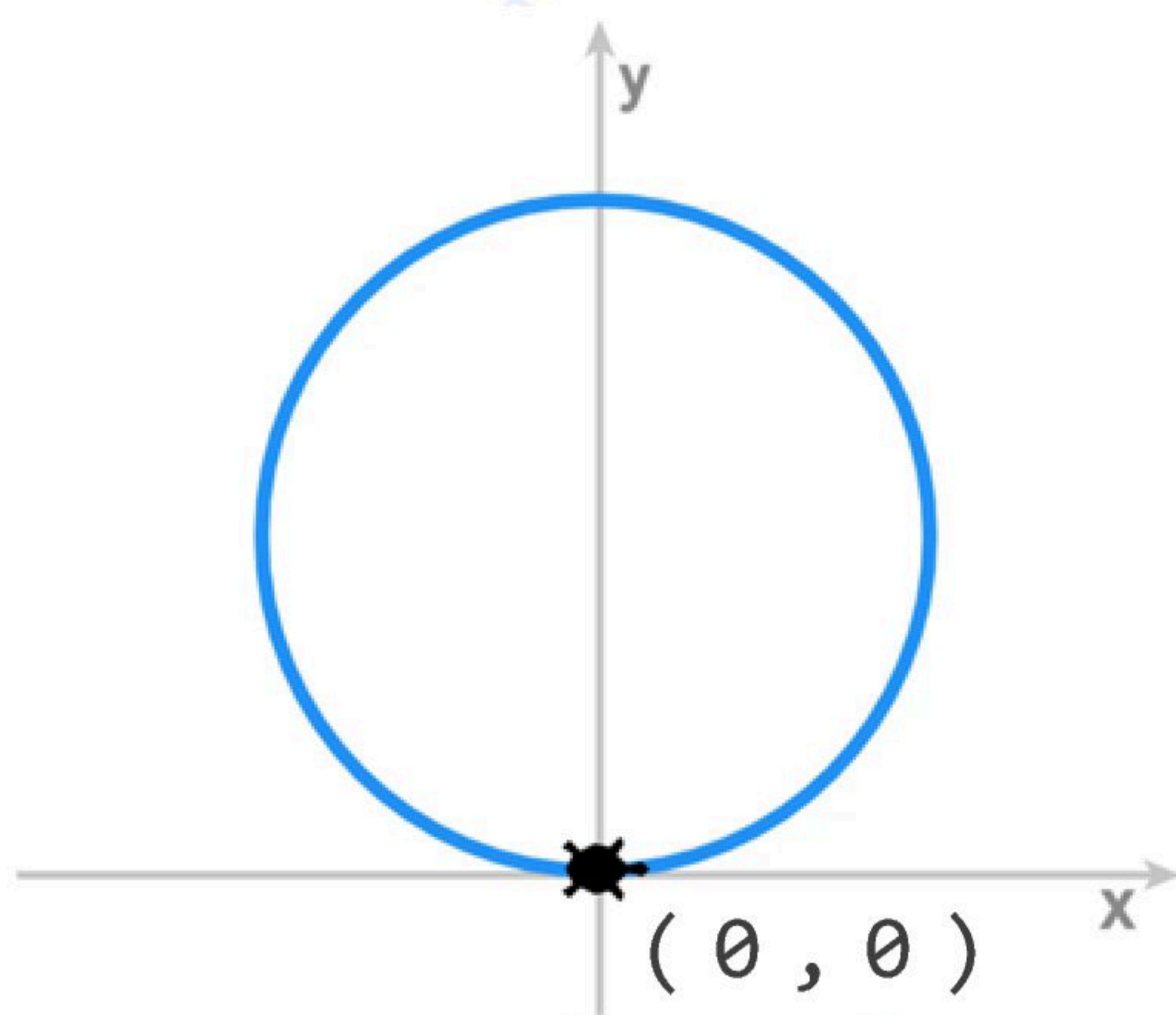


多边形

绘制一个指定半径的圆 / 圆弧 / 正多边形

radius参数设置

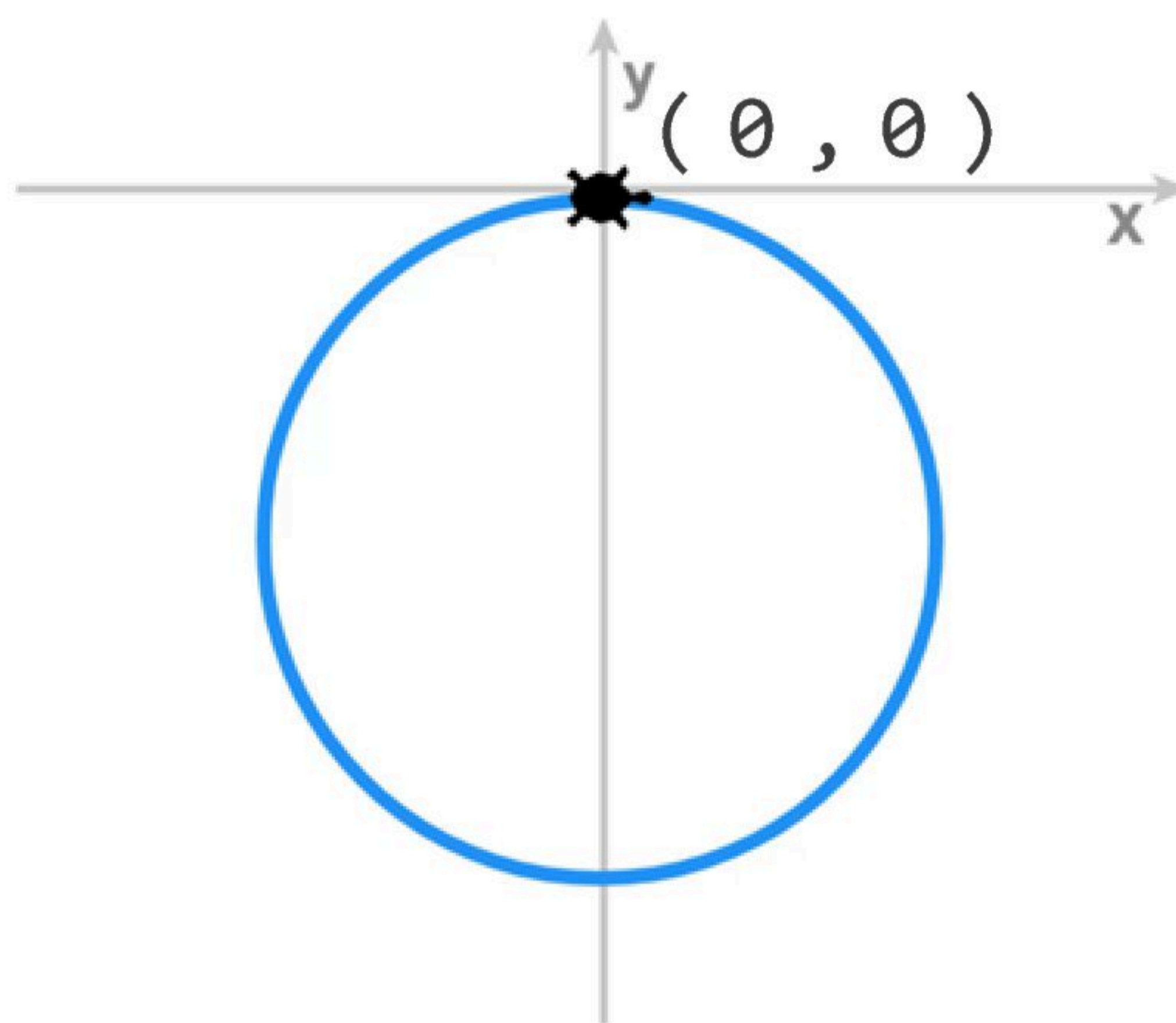
```
turtle.circle(100, None, None)
```



radius正数:

圆心在当前位置(小海龟)的左侧

```
turtle.circle(-100, None, None)
```



radius负数:

圆心在当前位置(小海龟)的右侧

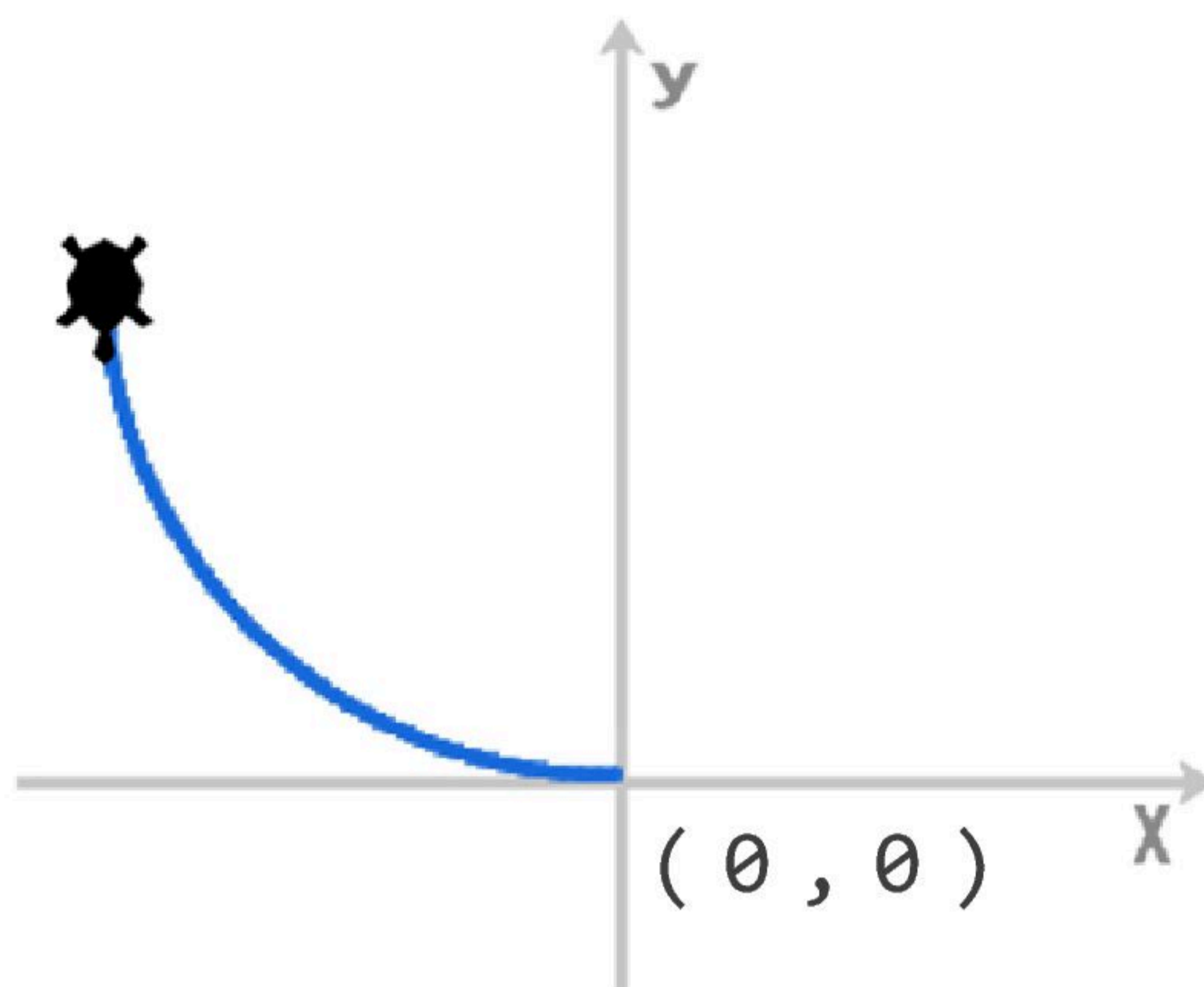
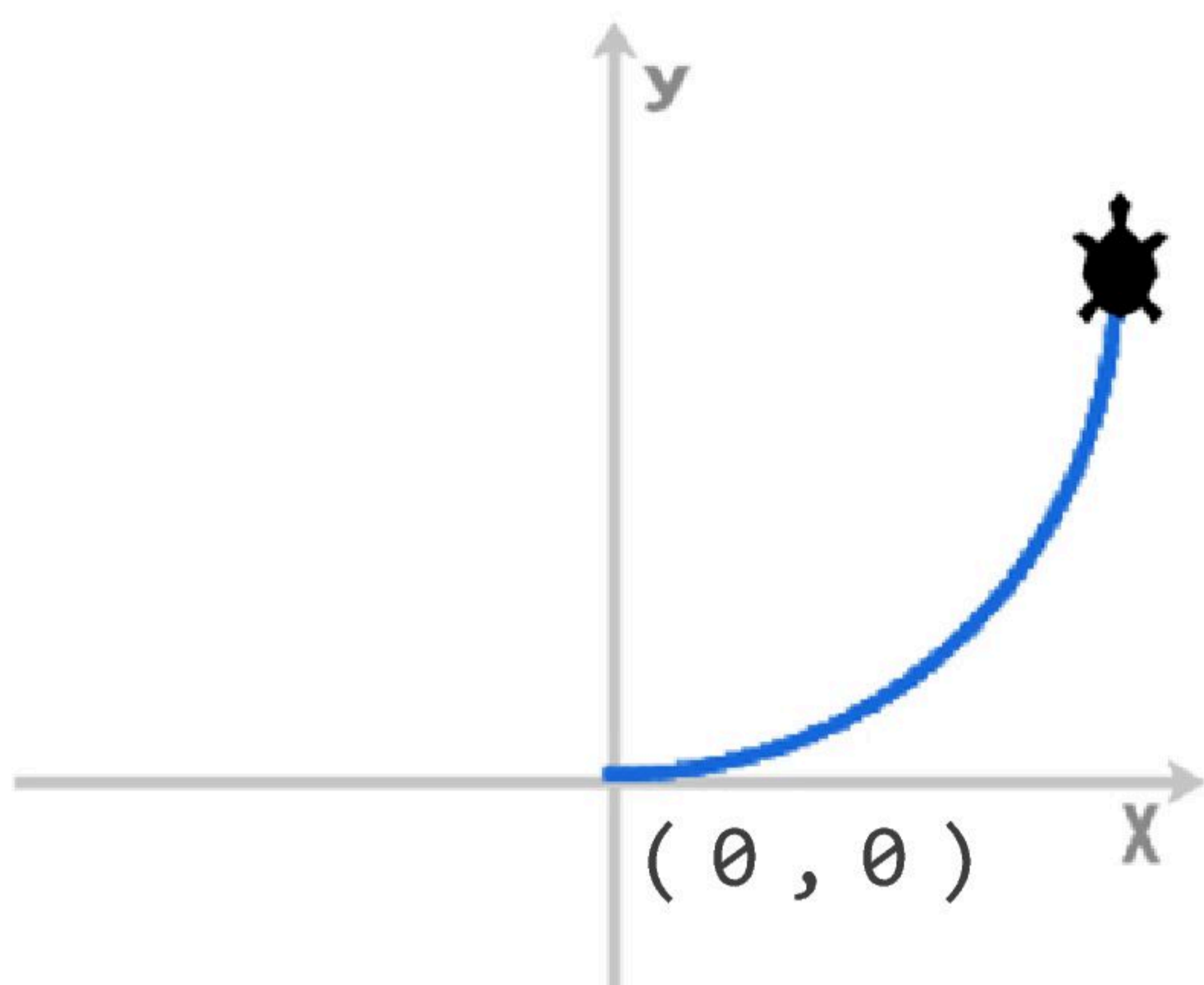
1.1

circle用法进阶

extent参数设置（半径为正）

```
turtle.circle(100, 90, None)
```

```
turtle.circle(100, -90, None)
```



extent正数：

顺着小海龟当前方向绘制

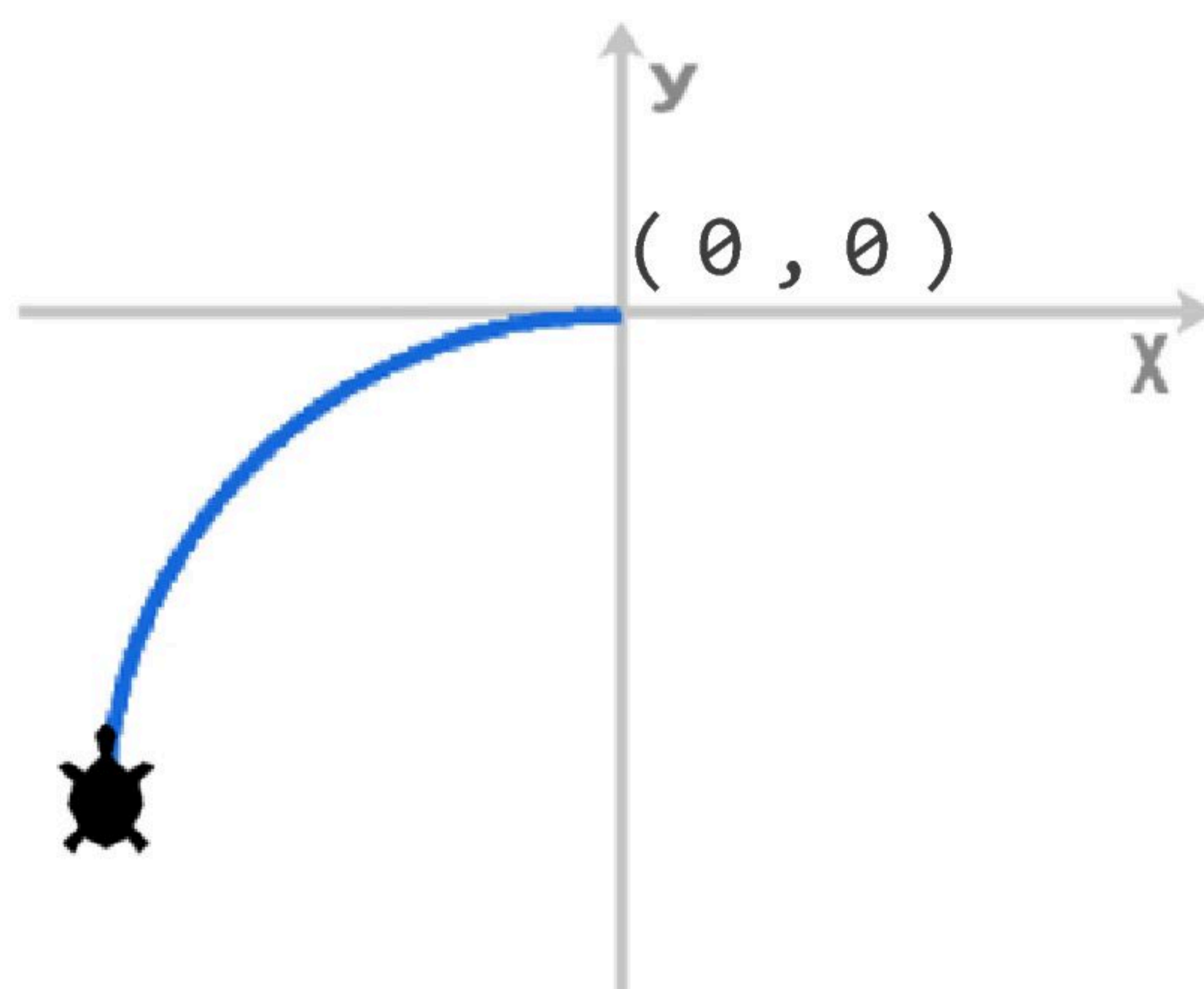
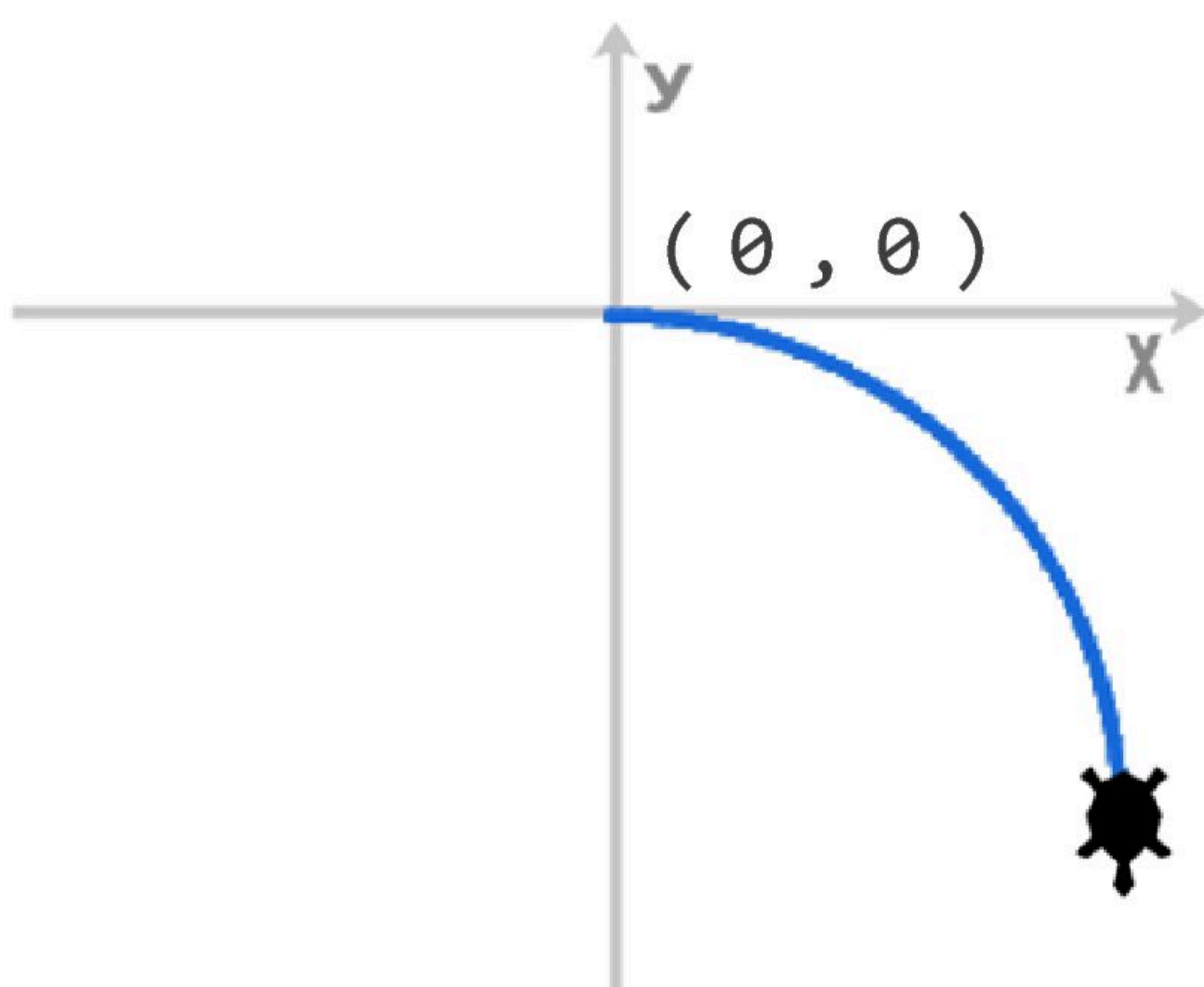
extent负数：

逆着小海龟当前方向绘制

extent参数设置（半径为负）

```
turtle.circle(-100, 90, None)
```

```
turtle.circle(-100, -90, None)
```



extent正数：

顺着小海龟当前方向绘制

extent负数：

逆着小海龟当前方向绘制

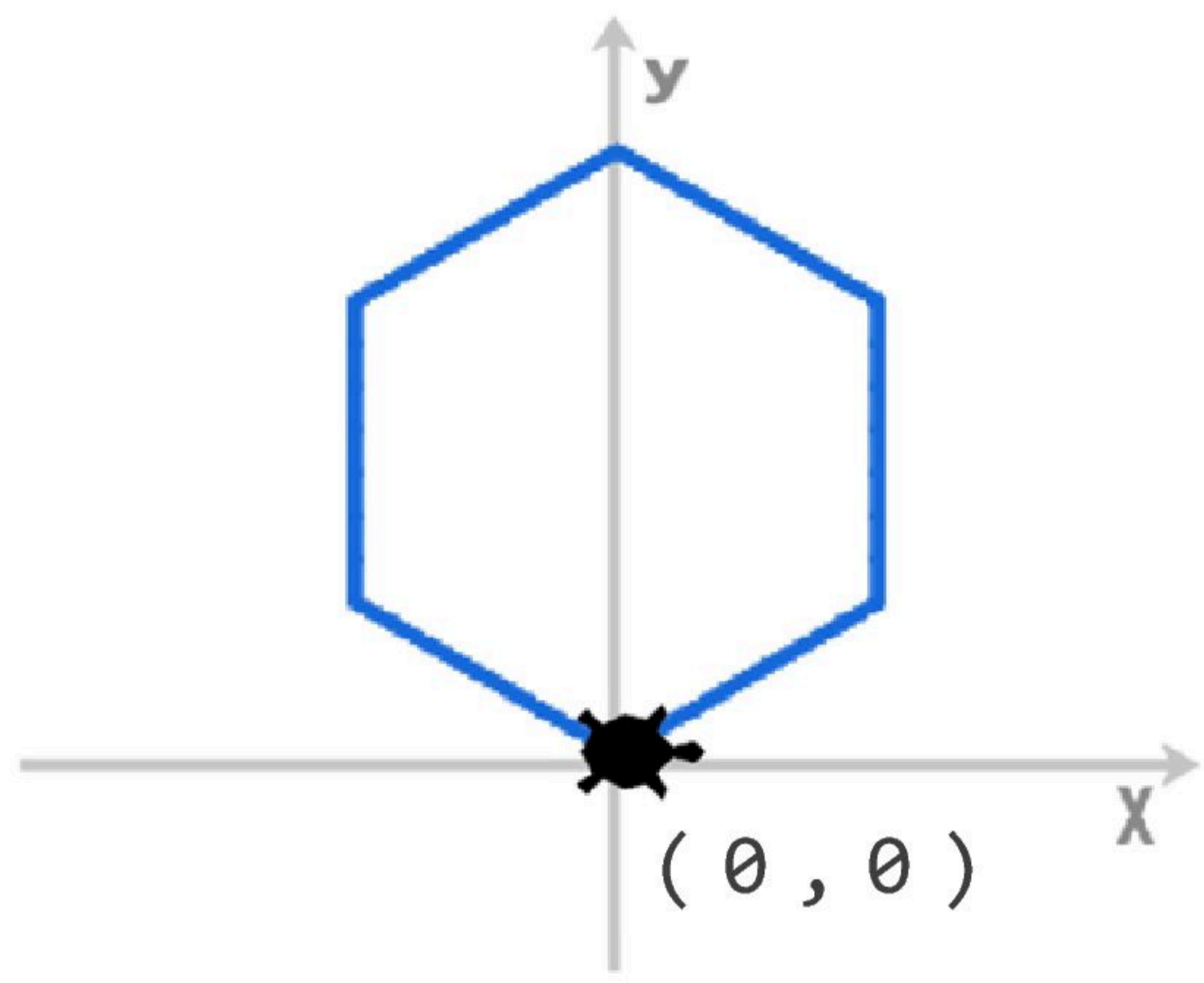
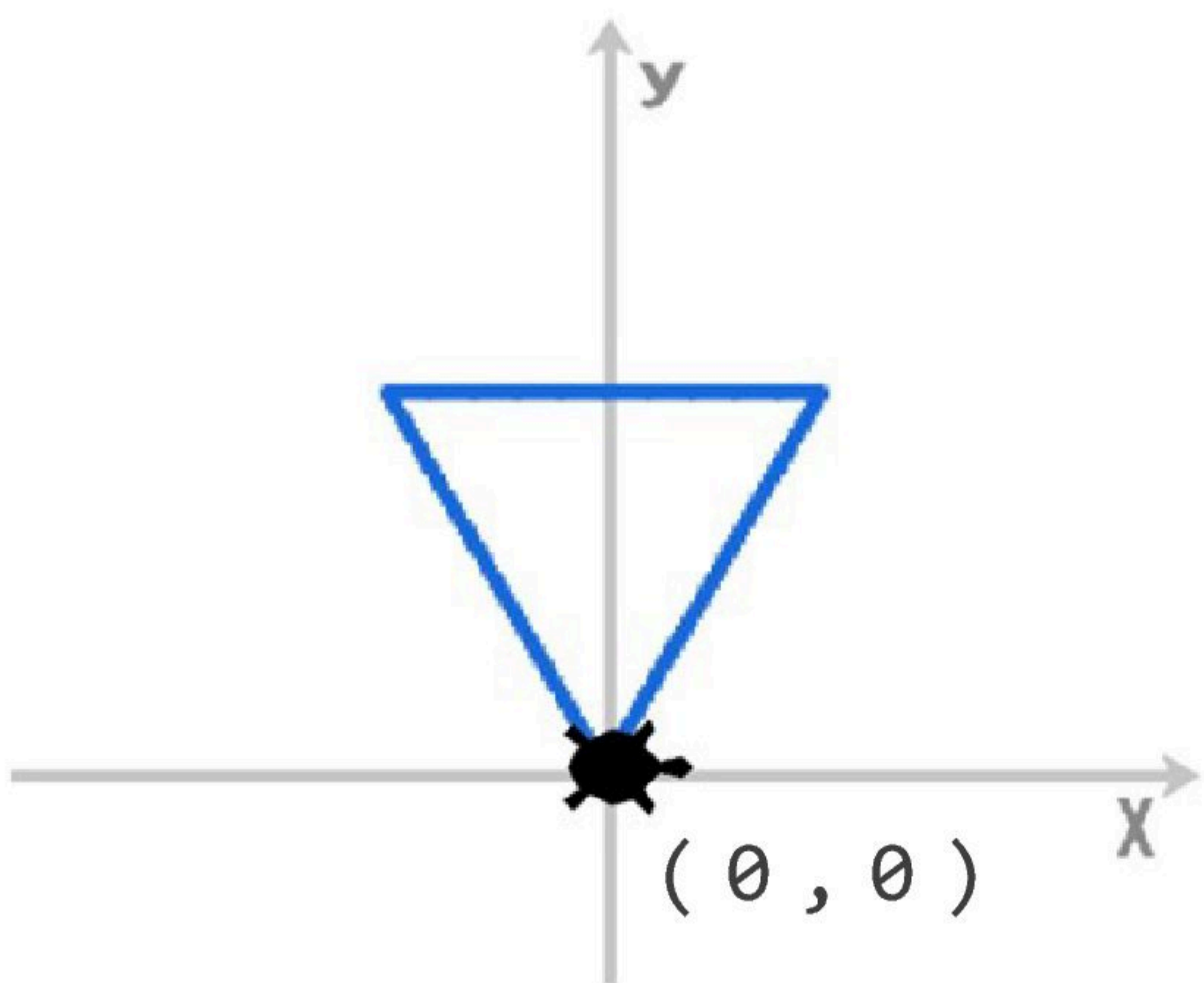
1.1

circle用法进阶

radius	extent	描述
> 0	> 0	圆心在小海龟的左侧顺着小海龟当前方向绘制
> 0	< 0	圆心在小海龟的左侧逆着小海龟当前方向绘制
< 0	> 0	圆心在小海龟的右侧顺着小海龟当前方向绘制
< 0	< 0	圆心在小海龟的右侧逆着小海龟当前方向绘制

steps 参数设置

`turtle.circle(100, steps=3)` `turtle.circle(100, steps=6)`



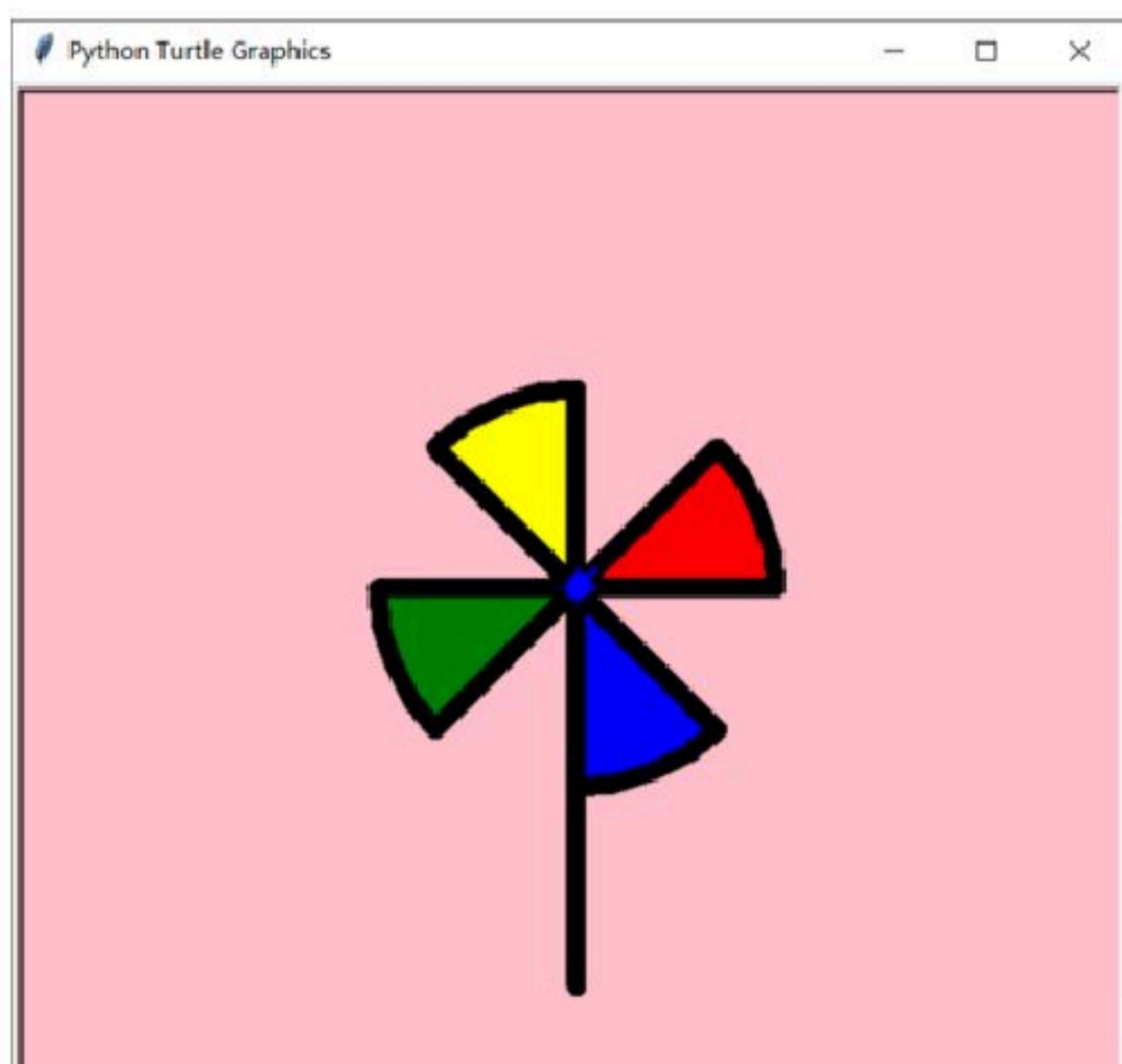
extent一般默认为 360°



想想试试: `turtle.circle(-100,90)` 会如何画弧?

1.2

色彩盛宴



`turtle.bgcolor('pink')` 设置或返回背景颜色

`turtle.pencolor('black')` 设置或返回画笔颜色

`turtle.fillcolor('green')` 设置或返回填充颜色

`turtle.color(画笔颜色, 填充颜色)` **==** `turtle.pencolor()`
`turtle.fillcolor()`

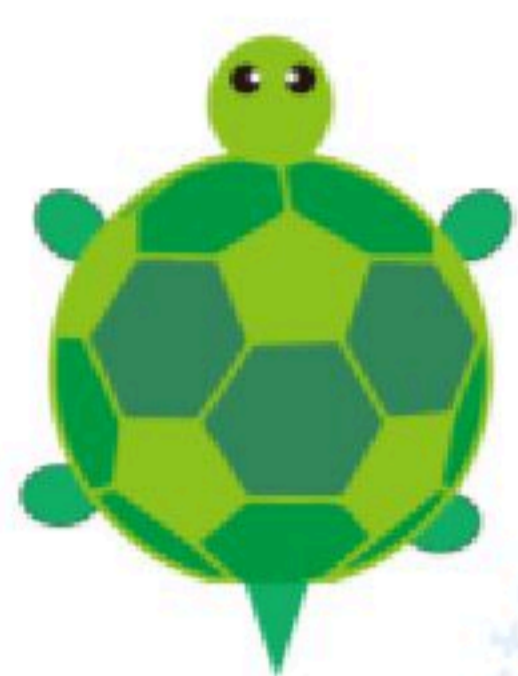
要填充的颜色设定好了，那怎么知道要填充哪部分呢？



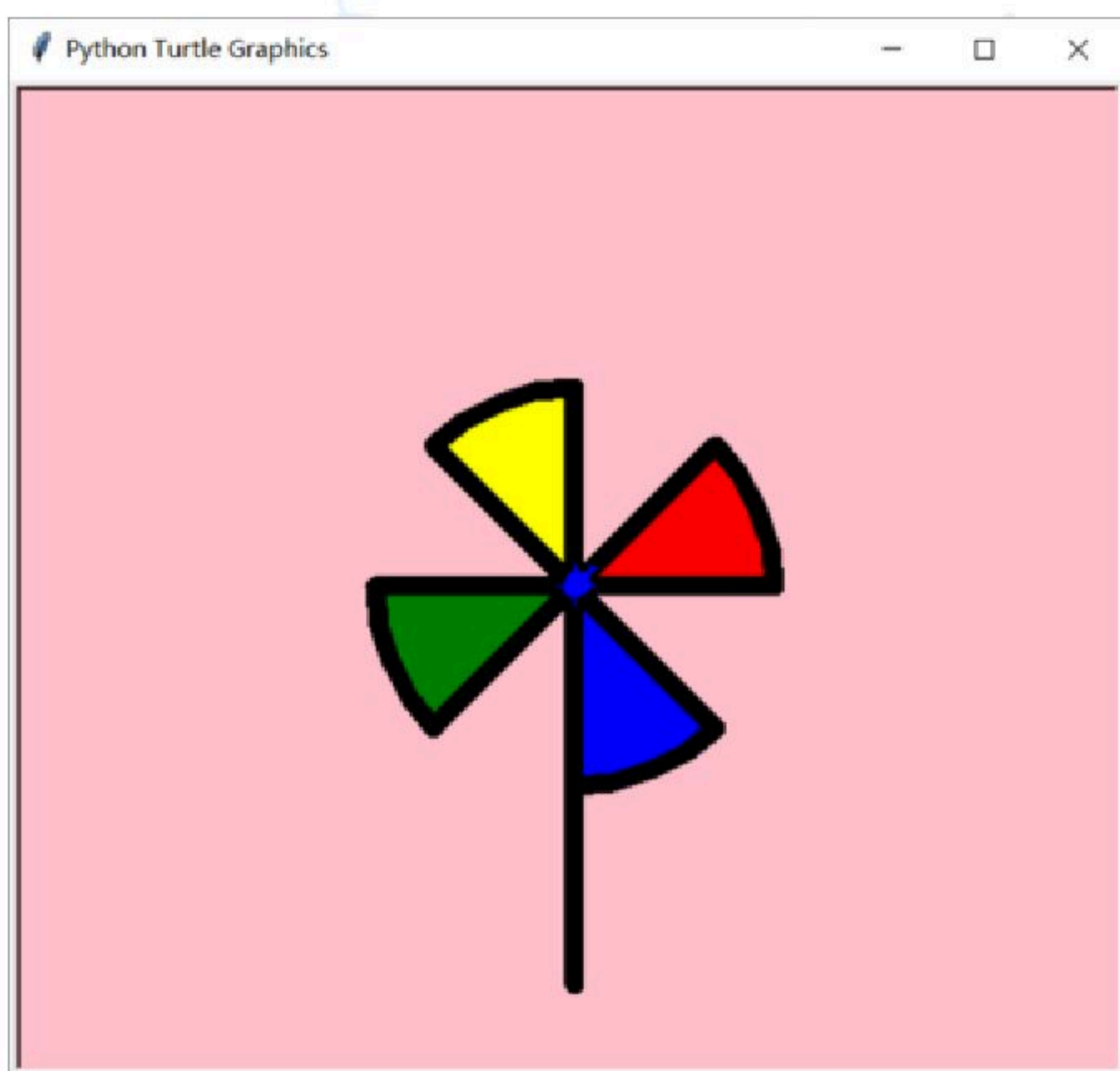
```
turtle.begin_fill () # 开始填充
...
<要被填充的部分>
...
turtle.end_fill() # 结束填充
```

1.3

终极考验

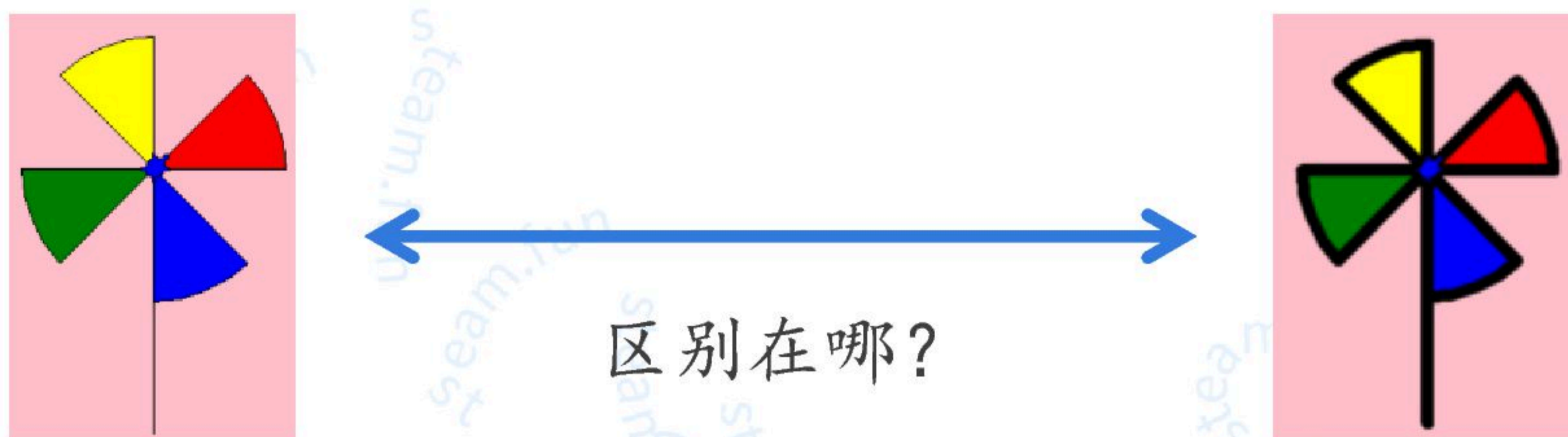


恭喜你来到这里，只要通过海龟岛的终极考验，就可以开启新的旅程！
考验开始！小小勇士们，去完成“多彩风车”的作品吧



- 步骤拆解：
1. 初始设置
 2. 画竖线
 3. 画红色扇叶
 4. 画黄色扇叶
 5. 画绿色扇叶
 6. 画蓝色扇叶

1. 初始设置



如何设置画笔的粗细?

- `turtle.pensize()`/`turtle.width()`
 钢笔尺寸 宽度

初始设置

```
import turtle as t
```

```
t.shape('turtle') # 海龟形状
```

```
t.bgcolor('pink') # 粉色背景
```

```
t.pencolor('black') # 黑色画笔
```

```
t.pensize(10) # 画笔宽度为10
```

2. 画竖线

```
# 移到(0, -200), 不留下痕迹
```

```
t.penup()
```

```
t.goto(0, -200)
```

```
t.pendown()
```

```
# 画竖线
```

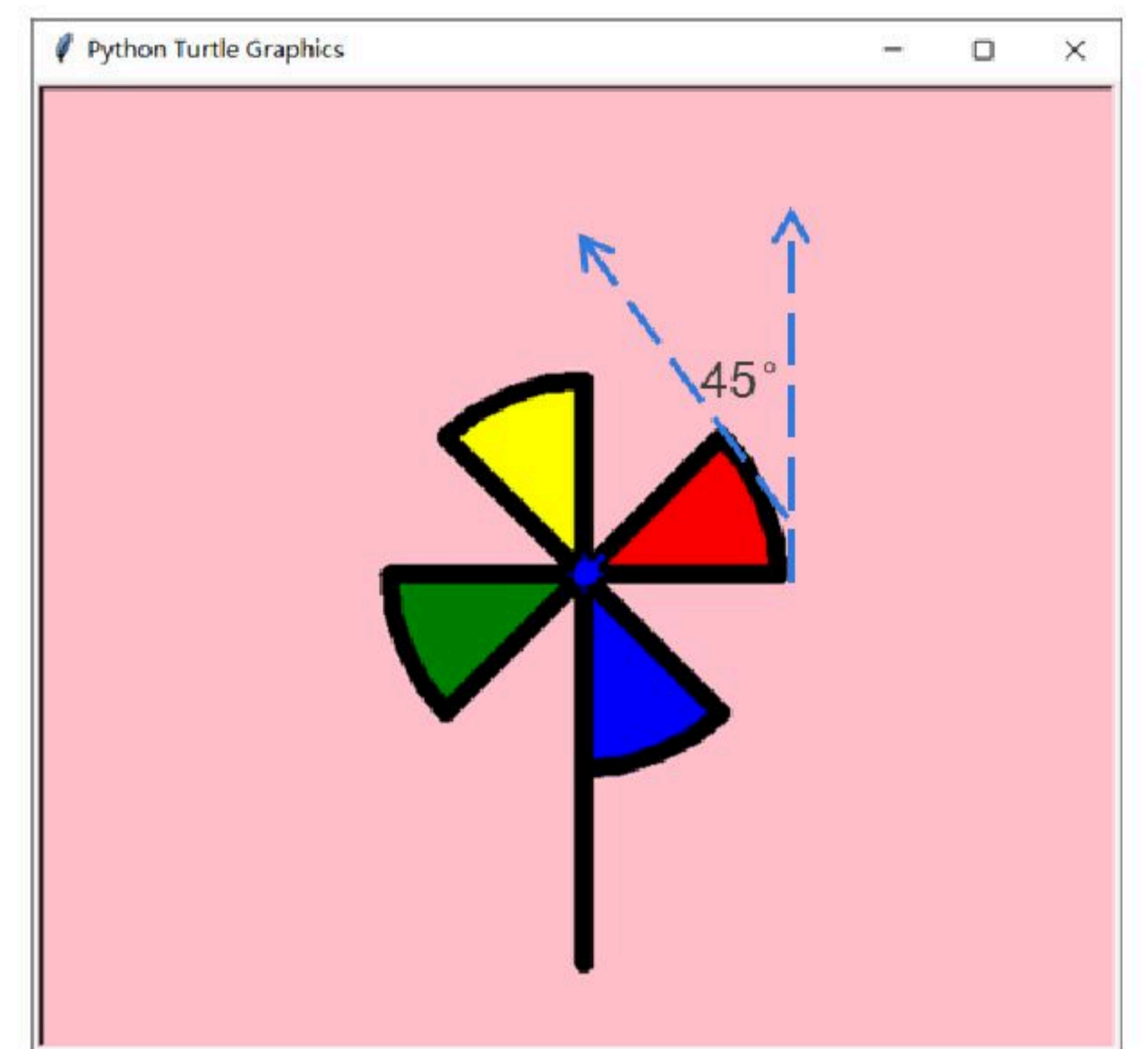
```
t.goto(0,0)
```

3. 画红色扇叶

```

# 画红色扇叶
t.fillcolor('red') # 红色填充
t.begin_fill() # 开始填充
t.seth(0) # 朝向右
t.forward(100) # 前进100像素
t.left(90) # 左转90°, 朝向上
t.circle(100,45) # 画扇叶圆弧, 角度45
t.goto(0,0) # 回到原点
t.end_fill() # 结束填充

```

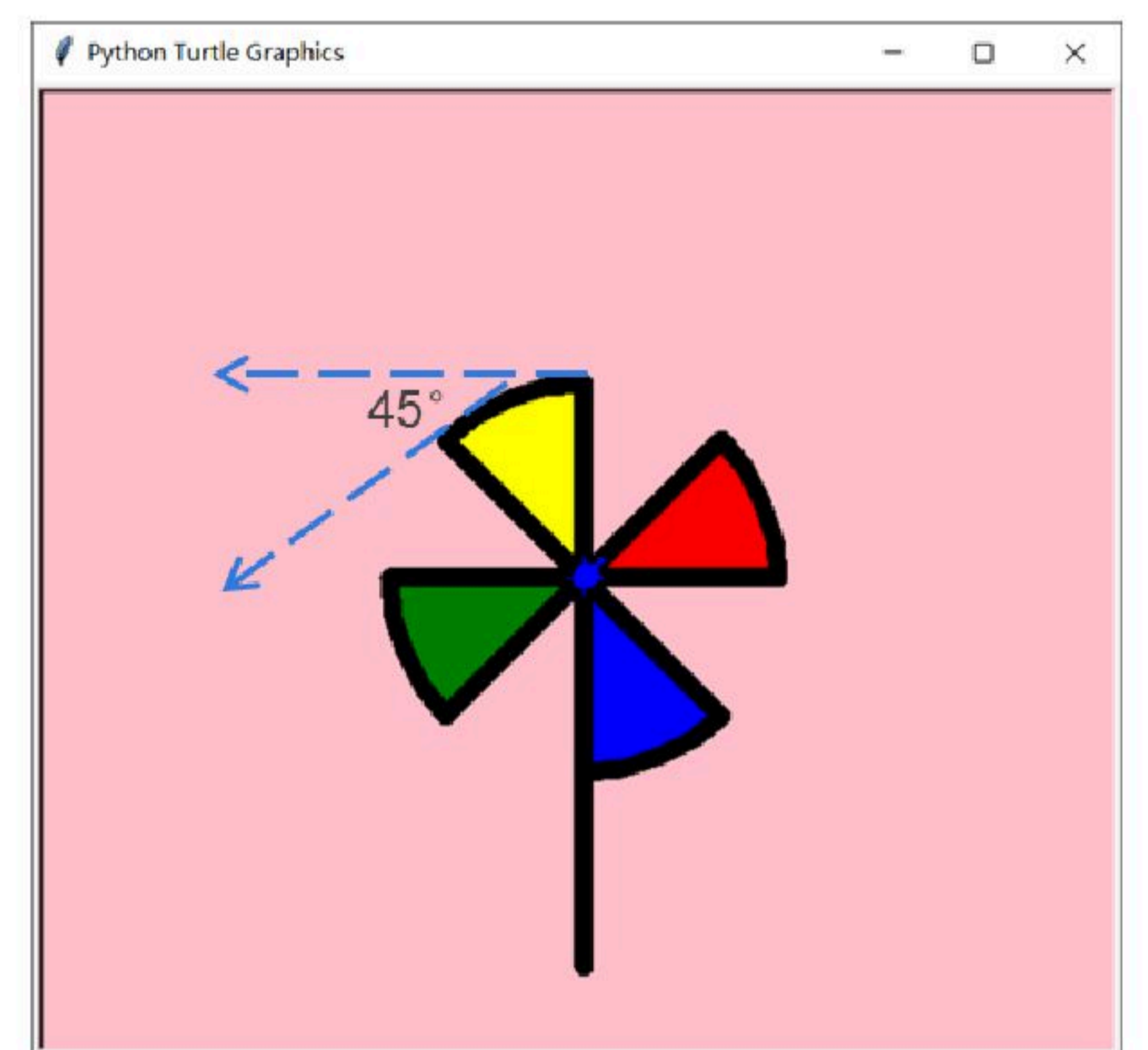


4. 画黄色扇叶

```

# 画黄色扇叶
t.fillcolor('yellow') # 黄色填充
t.begin_fill() # 开始填充
t.seth(90) # 朝向上
t.forward(100) # 前进100像素
t.left(90) # 左转90°, 朝向左
t.circle(100,45) # 画扇叶圆弧, 角度45
t.goto(0,0) # 回到原点
t.end_fill() # 结束填充

```



5. 画绿色扇叶

6. 画蓝色扇叶

试着自己完成绿色扇叶、蓝色扇叶的创作吧!



2. 强化练习

1. 下列哪个选项的代码，可以画出一个直径（注意是直径，不是半径）为100的半圆？（ ）

A. turtle.circle(100,360)

B. turtle.circle(50,180)

C. turtle.circle(100,180)

D. turtle.circle(50,360)

2. 执行下列turtle库的指令后，画笔最终为哪种颜色？（ ）

A. 黄色

B. 红色

C. 蓝色

D. 绿色

```
import turtle
turtle.pencolor("blue")
turtle.color("red","green")
turtle.fillcolor("yellow")
```

3. 已知以下指令，想要绘制一个画笔粗细为5，颜色为红色，半径为50的圆，请问正确的排序方式是？（ ）

① turtle.pencolor("red")

② import turtle

③ turtle.pensize(5)

④ turtle.circle(50)

A. ①②③④

B. ①③②④

C. ②③①④

D. ③④①②



2. 强化练习

4. 先后执行Python代码 `import turtle` 和 `turtle.circle(10,steps=3)` 后, 正确的说法是 ()

- A. 将绘制出半径为10, 弧度为3的圆
- B. 将绘制出半径为10, 角度为3的圆
- C. 将绘制出边长为10的正三角形, 即等边三角形
- D. 将绘制出半径为10的内接圆正三角形, 即等边三角形

5. `turtle.color('red','yellow')`命令中定义的颜色分别为? ()

- A. 背景为黄色, 画笔为红色
- B. 背景为红色, 画笔为黄色
- C. 画笔为红色, 填充为黄色
- D. 画笔为黄色, 填充为红色

3. 术语箱

bgcolor 背景颜色

pencolor 画笔颜色

fillcolor 填充颜色

begin_fill 开始填充

end_fill 结束填充

4. 课后挑战

多彩机器人

在上节课作业的基础上进行创作

要求如下：

- (1) 设置画笔粗细为5，颜色为默认黑色
- (2) 背景为橙色 "orange"
- (3) 正方形蓝脸：填充色为蓝色 "blue"
- (4) 三角形红嘴：填充色为红色 "red"

