Chapter 4

Dr. Raza Ul Mustafa Khokhar
American University
Computer Science Department - CSC-148

Turtle example

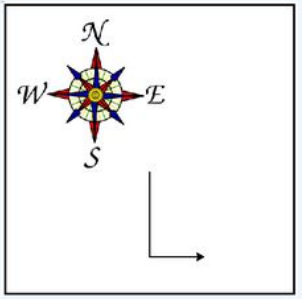
- In this chapter we will introduce a module that allows us to create a data object called a turtle that can be used to draw pictures

- Wait: There are other modules and functionality also there, for example numpy for mathematical calculations

Example

- Turtle graphics, as it is known, is based on a very simple metaphor.
- Imagine that you have a turtle that understands English. You can tell your turtle to do simple commands such as go forward and turn right.
- **As the turtle moves around, if its tail is down touching the ground,** it will draw a line (leave a trail behind) as it moves.
- If you tell your turtle to lift up its tail it can still move around but will not leave a trail. As you will see, you can make some pretty amazing drawings with this simple capability.

Few functions




```
import turtle                 # allows us to use the turtles library
wn = turtle.Screen()          # creates a graphics window
alex = turtle.Turtle()        # create a turtle named alex
alex.forward(150)             # tell alex to move forward by 150 units
alex.left(90)                 # turn by 90 degrees
alex.forward(75)              # complete the second side of a rectangle
```

1. The first line tells Python to load a **module** named turtle
2. Remember that Python is case sensitive, so the module name, turtle, with a lowercase t, is different from the type Turtle because of the uppercase T.

Few more functionality

```
import turtle

wn = turtle.Screen()
wn.bgcolor("lightgreen")        # set the window background color

tess = turtle.Turtle()
tess.color("blue")              # make tess blue
tess.pensize(3)                 # set the width of her pen

tess.forward(50)
tess.left(120)
tess.forward(50)

wn.exitonclick()  # wait for a user click on the canvas
```

```
import turtle
wn = turtle.Screen()             # Set up the window and its attributes
wn.bgcolor("lightgreen")
tess = turtle.Turtle()           # create tess and set some attributes
tess.color("hotpink")
tess.pensize(5)
alex = turtle.Turtle()           # create alex
tess.forward(80)                 # Let tess draw an equilateral triangle
tess.left(120)
tess.forward(80)
tess.left(120)
tess.forward(80)
tess.left(120)                   # complete the triangle
tess.right(180)                  # turn tess around
tess.forward(80)                 # move her away from the origin
alex.forward(50)                 # make alex draw a square
alex.left(90)
alex.forward(50)
alex.left(90)
alex.forward(50)
alex.left(90)
alex.forward(50)
alex.left(90)
wn.exitonclick()
```

Output of previous code...

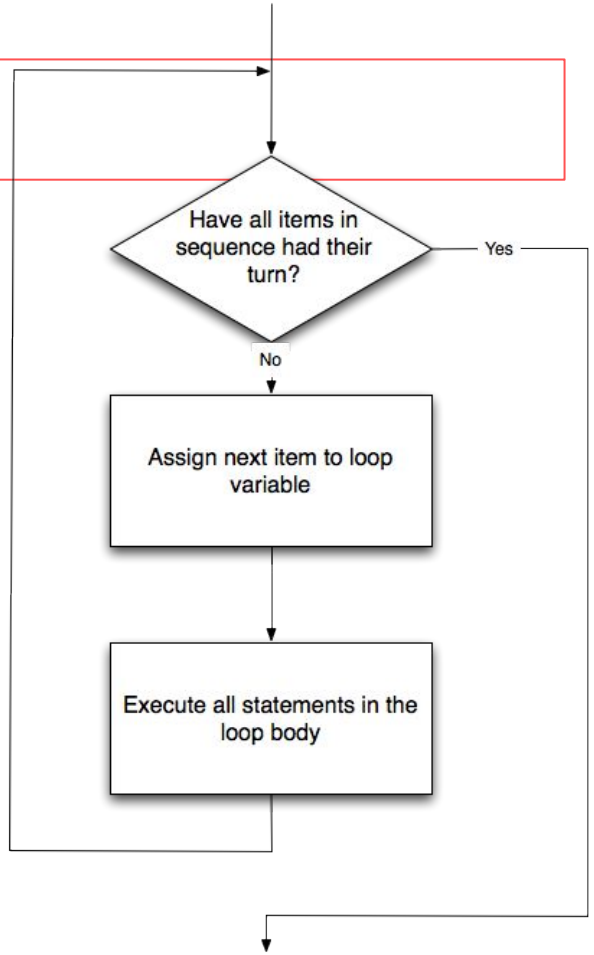
We will run live to see how it works

The for loop

- A basic building block of all programs is to be able to repeat some code over and over again. In computer science, we refer to this repetitive idea as iteration.

```
for name in ["Joe", "Amy", "Brad", "Angelina", "Zuki", "Thandi", "Paris"]:
    print("Hi", name, "Please come to my party on Saturday!")
```

Flow of for loop



Turtle example

```
import turtle             # set up alex
wn = turtle.Screen()
alex = turtle.Turtle()

for i in [0, 1, 2, 3]:      # repeat four times
    alex.forward(50)
    alex.left(90)

wn.exitonclick()
```

Change the color

```
import turtle             # set up alex
wn = turtle.Screen()
alex = turtle.Turtle()

for i in [yellow, red, green, blue]:      # repeat four times
    alex.forward(50)
    alex.left(90)

wn.exitonclick()
```

Let see how it looks now!

```
import turtle              # set up alex
wn = turtle.Screen()
alex = turtle.Turtle()

for aColor in ["yellow", "red", "purple", "blue"]:
    alex.color(aColor)
    alex.forward(50)
    alex.left(90)

wn.exitonclick()
```

The range function

- This is how we doing before? How to use range function here?

```
import turtle             # set up alex
wn = turtle.Screen()
alex = turtle.Turtle()

for i in [0, 1, 2, 3]:    # repeat four times
    alex.forward(50)
    alex.left(90)

wn.exitonclick()
```

Range function in action!

```
for i in range(4):
    # Executes the body with i = 0, then 1, then 2, then 3
for x in range(10):
    # sets x to each of ... [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```

More things to do with range!

- range(start, beyondLast, step)

```
print(list(range(0, 18, 2)))
print(list(range(0, 20, 2)))
print(list(range(10, 0, -1)))
```

More on range... Turtle

```
import turtle
wn = turtle.Screen()
wn.bgcolor("lightgreen")
tess = turtle.Turtle()
tess.color("blue")
tess.shape("turtle")

print(list(range(5, 60, 2)))
tess.up()                       # this is new
for size in range(5, 60, 2):      # start with size = 5 and grow by 2
    tess.stamp()                  # leave an impression on the canvas
    tess.forward(size)            # move tess along
    tess.right(24)                # and turn her

wn.exitonclick()
```

Many more methods of Turtle use-case

- Less priorities, More on syntax and logic building, OK!

https://shorturl.at/fIJW4

Let’s start little more practice … which is important

- Open your IDE

Let’s start playing with Nested loops

Few shapes

Dry run … Very important ! ! !