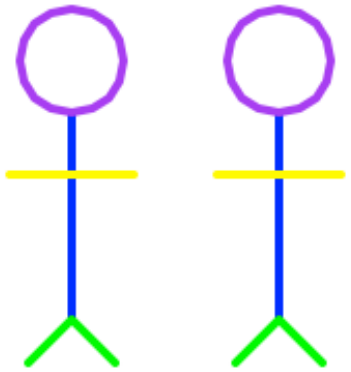
CSCI-141 Computer Science 1

Recitation

01
Python Introduction
Functions
Turtle

Stick Figures

- **The goal is to draw these stick figures using Python and the turtle module**

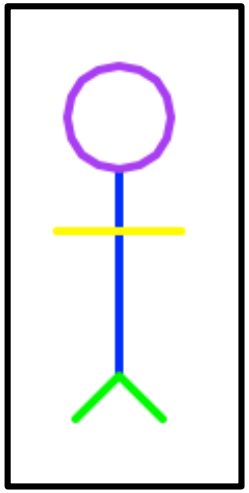


Stick Figure Drawing Algorithm

- **Recall from lecture that an *algorithm* is a special set of instructions**
- **An algorithm for drawing a single stick figure could be composed of the following *tasks***
  - Draw the torso
  - Draw the legs
  - Draw the arms
  - Draw the head

Functions

- **We can assign each task to a separate *function* in Python**



```
def draw_stick_figure():
    draw_torso()
    draw_legs()
    draw_arms()
    draw_head()
```

Turtle

- **The turtle package has functions in it for drawing our stick figures**

```
import turtle
```

- **Recall the default settings for the turtle:**
  - The turtle is at the center of the canvas
  - The turtle is down on the canvas
  - The turtle is facing east (0 º)

Turtle Functions

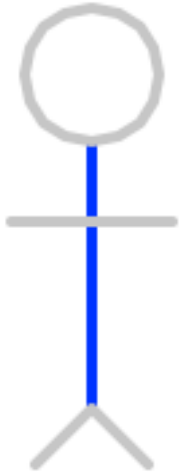
- **Recall the turtle drawing functions**
  - `turtle.up()`
  - `turtle.down()`
  - `turtle.left(`*`angle`*`)`
  - `turtle.right(`*`angle`*`)`
  - `turtle.forward(`*`distance`*`)`
  - `turtle.back(`*`distance`*`)`
  - `turtle.circle(`*`radius`*`)`

Preconditions and Postconditions

- **Let's define the initial state of the turtle before (preconditions) and after (postconditions) a drawing function is invoked**
  - The turtle is located at the top of the torso and bottom center of the head
  - The turtle is up
  - The turtle is facing east

Draw Torso

- **Here's the complete function `draw_torso`, for drawing the torso while adhering to the pre and post-conditions**



```
def draw_torso():
    turtle.right(90)
    turtle.down()
    turtle.forward(100)
    turtle.up()
    turtle.back(100)
    turtle.left(90)
```

Draw Legs

- **The complete function `draw_legs`**

```
def draw_legs():
    turtle.right(90)
    turtle.forward(100)
    turtle.right(45)
    turtle.down()
    turtle.forward(30)
    turtle.back(30)
    turtle.left(90)
    turtle.forward(30)
    turtle.back(30)
    turtle.right(45)
    turtle.up()
    turtle.back(100)
    turtle.left(90)
```

Draw Arms

- **The complete function `draw_arms`**

```
def draw_arms():
    turtle.right(90)
    turtle.forward(30)
    turtle.left(90)
    turtle.down()
    turtle.forward(30)
    turtle.back(30)
    turtle.left(180)
    turtle.forward(30)
    turtle.back(30)
    turtle.left(90)
    turtle.up()
    turtle.back(30)
    turtle.left(90)
```

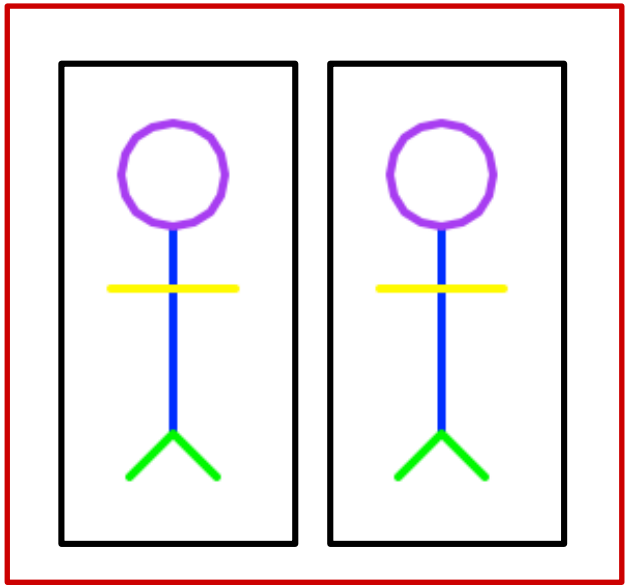
Draw Head

- **The complete function `draw_head`**

```
def draw_head():
    turtle.down()
    turtle.circle(25)
    turtle.up()
```

Function Reuse

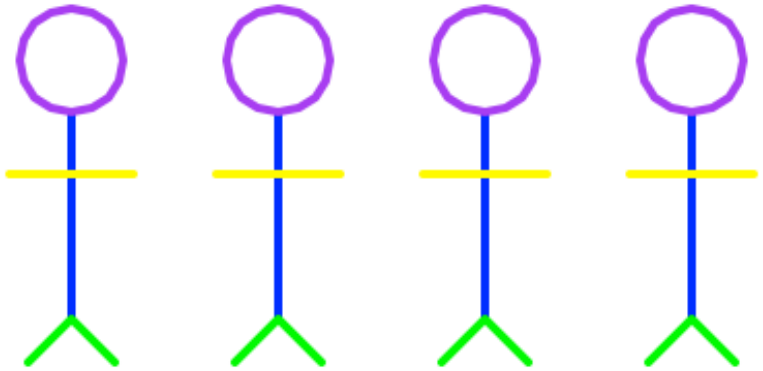
- To draw the entire image, we can have a ***main function*** that reuses `draw_stick_figure`



```
def main():
    draw_stick_figure()
    turtle.forward(100)
    draw_stick_figure()
```

Function Reuse

- **By reusing `draw_stick_figure` we can draw any number of them easily**



Recitation Code

- **Stick Figures**
  - https://www.cs.rit.edu/~csci141/Recitations/01/Code/stick_figures.py