Smith College
Computer Science

Lecture Notes
CSC111

Week 7 — Spring 2018

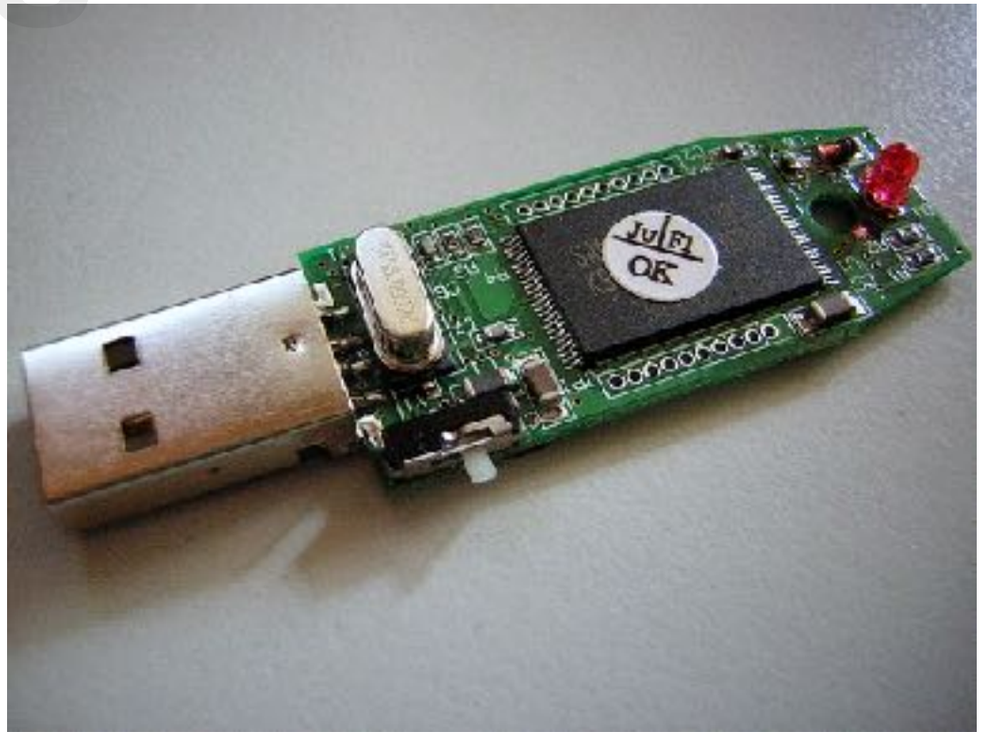
Dominique Thiébaut
dthiebaut@smith.edu

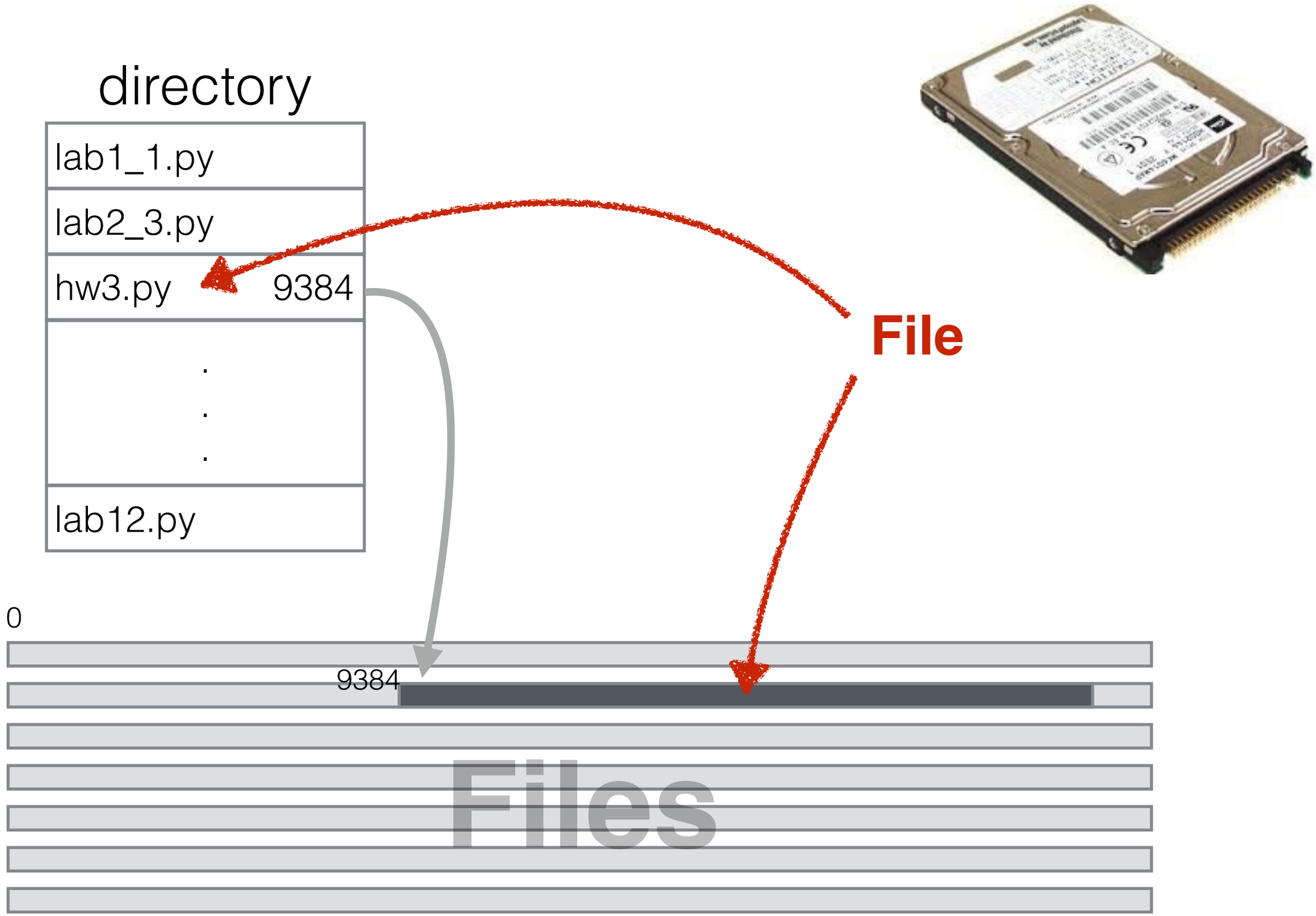
Midterm Grades available later today (3/19/18)

Outline

- **A Second Look at Files**
  - **Reading Files**
  - **Writing Files**
- **Graphics (*Chapter 4*)**
- **If Statements (*Chapter 7*)**
- **Animation**

Files




directory
lab1_1.py
lab2_3.py
hw3.py
9384
lab12.py
File
0
9384
Files

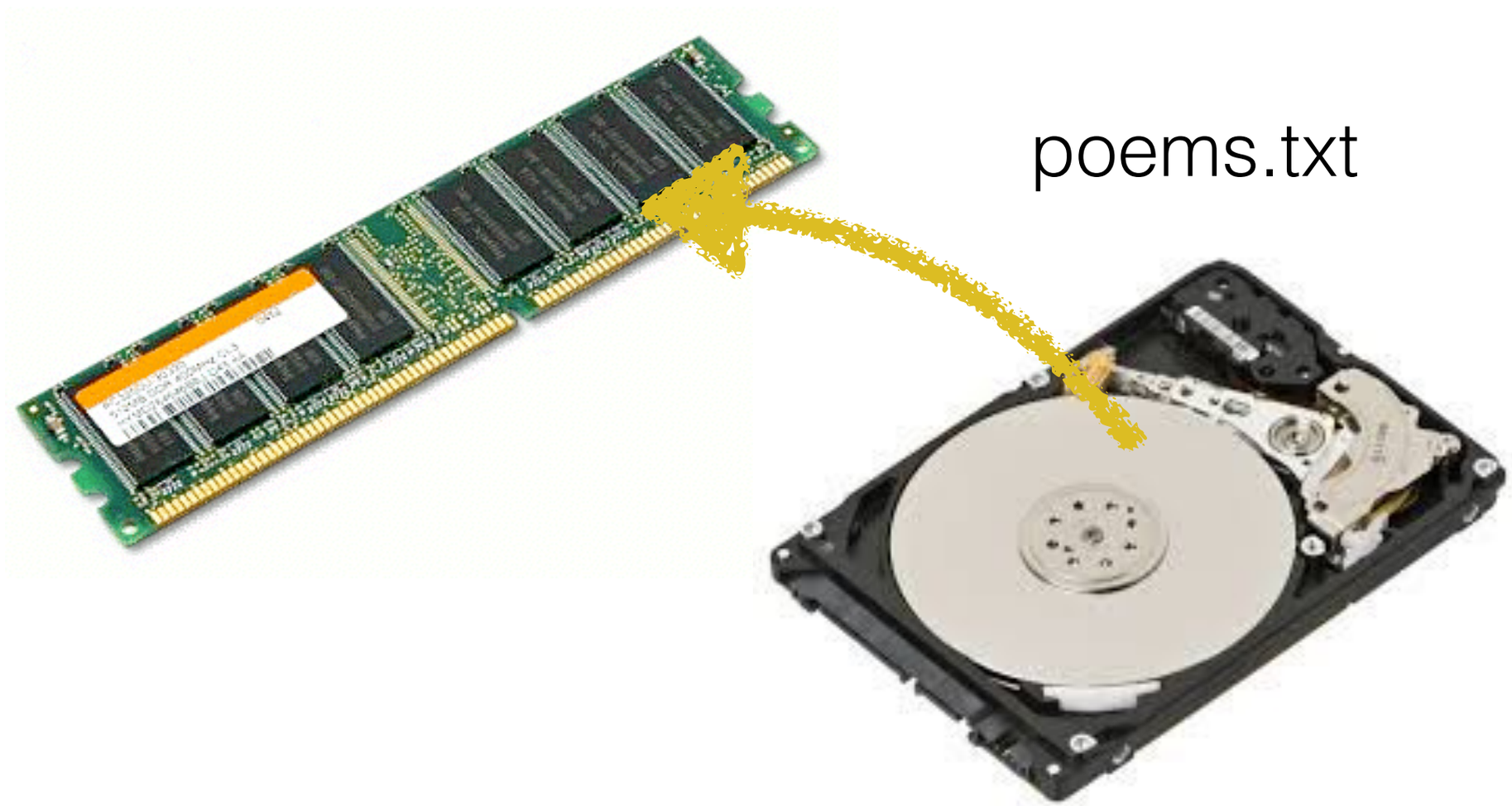
Outline

- A Second Look at Files
  - Reading Files
  - Writing Files
- Graphics
- If Statements
- Animation

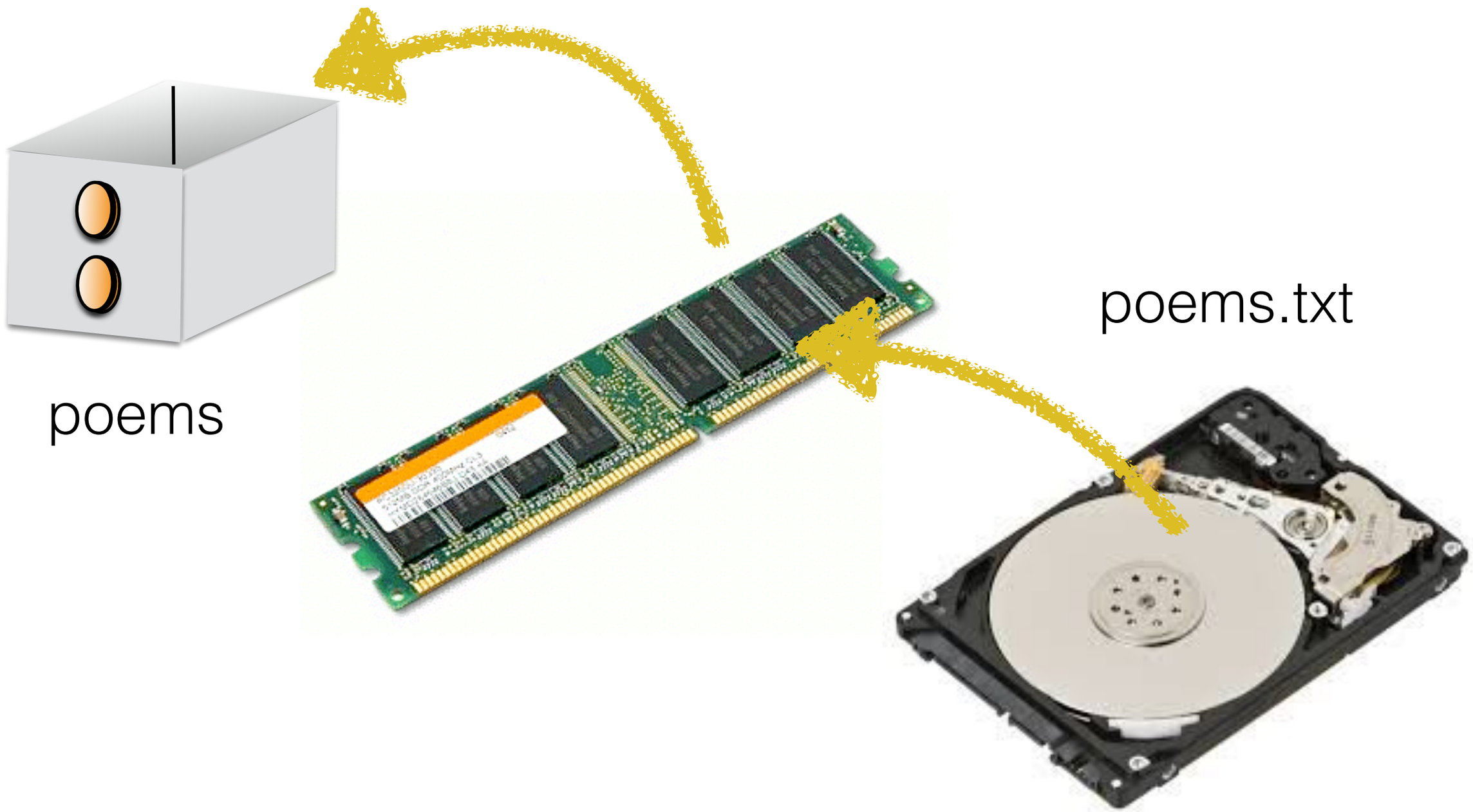
Opening a File for **Reading**

poems.txt

Opening a File for **Reading**



Opening a File for **Reading**



Reading a file:

3 Steps:

- **Open**
- **Read**
- **Close**

```
# read the file into a single string
file1 = open( "poems.txt", "r" )
text = file1.read()
file1.close()
```

```
# read the file into a single string
file1 = open( "poems.txt", "r" )
text = file1.read()
file1.close()
```

String

read()

```
# read the file into a single string
file1 = open( "poems.txt", "r" )
text = file1.read()
file1.close()
```

String

read()

```
# read the file into a list of strings
file2 = open( "poems.txt", "r" )
lines = file2.readlines()
file2.close()
```

```
# read the file into a single string
file1 = open( "poems.txt", "r" )
text = file1.read()
file1.close()
```

String

read()

```
# read the file into a list of strings
file2 = open( "poems.txt", "r" )
lines = file2.readlines()
file2.close()
```

List of String

readlines()

Python Shell

```
Python 3.1.1 (r311:74543, Aug 24 2009, 18:44:04)
[GCC 4.0.1 (Apple Inc. build 5493)] on darwin
Type "copyright", "credits" or "license()" for more information.
>>> demo ="""
```

Demo Time

```
"""
>>>

# Count the number of lines in a file containing
# email messages (emails.txt)
# Count the number of words in a file
# (facebookNews.txt)
```

Ln: 14 Col: 4

Important File Property

- When working with the same file several times in a program, make sure to close it before reading it ***again***.
- You ***cannot*** read a file twice without closing it between reads.

Outline

- A Second Look at Files
  - Reading Files
  - **Writing Files**
- Graphics
- If Statements
- Animation

Writing a file

- **Open**
- **Write**
- **Close**

```
# write a string to a file
text = """The quick red fox
jumped over the lazy brown dog"""

file3 = open( "poems2.txt", "w" )
file3.write( text )
file3.close()
```

Python Shell

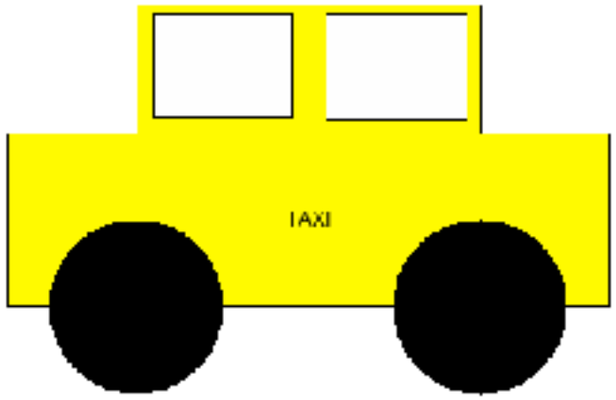
```
Python 3.1.1 (r311:74543, Aug 24 2009, 18:44:04)
[GCC 4.0.1 (Apple Inc. build 5493)] on darwin
Type "copyright", "credits" or "license()" for more information.
>>> demo ="""
```


```
"""
>>>
```

```
# write a personalized thank-you letter for
# donors to a campaign, and save each to file.
# [("Alex", 100), ("Joe", 10), ("Fifi", 50)]
```

writeThankYous.py

Computer Graphics (Covered in Chapter 4)

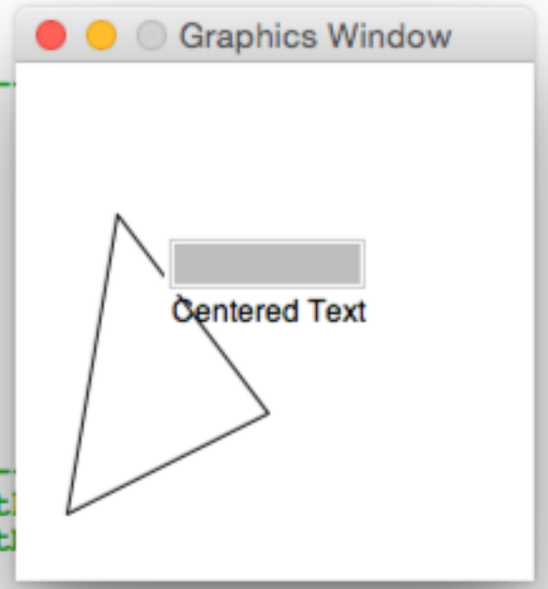


Graphics Library

- Can be found here: http://cs.smith.edu/dftwiki/index.php/Zelle%27s_Graphics.py_for_Python_3
- And in the Links and Resources section of the class **Web page**

graphics.py - /Users/thiebaut/Desktop/Dropbox/111/Week7/graphics.py

```
# graphics.py
"""Simple object oriented graphics library

The library is designed to make it very easy for novice programmers to
experiment with computer graphics in an object oriented fashion. It is
written by John Zelle for use with the book "Python Programming: An
Introduction to Computer Science" (Franklin, Beedle & Associates).

LICENSE: This is open-source software released under the terms of the
GPL (http://www.gnu.org/licenses/gpl.html).

PLATFORMS: The package is a wrapper around Tkinter and should run on
any platform where Tkinter is available.

INSTALLATION: Put this file somewhere where Python can see it.

OVERVIEW: There are two kinds of objects in the library. The GraphWin
class implements a window where drawing can be done and various
GraphicsObjects are provided that can be drawn into a GraphWin. As a
simple example, here is a complete program to draw a circle of radius
10 centered in a 100x100 window:

--------------------------------------------------------
from graphics import *

def main():
    win = GraphWin("My Circle", 100, 100)
    c = Circle(Point(50,50), 10)
    c.draw(win)
    win.getMouse() # Pause to view result
    win.close()    # Close window when done

main()
--------------------------------------------------------
GraphWin objects support coordinate transformation t
setCoords method and pointer-based input through get

The library provides the following graphical objects:
    Point
    Line
```



Ln: 1 Col: 0

Doing Graphics:

1. **Open** a graphic window

2. **Draw** on it

3. **Close** it

4. Terminate the program

```python
from graphics import *

def main():
    win = GraphWin("CSC111", 600, 400)
    c = Circle(Point(50,50), 10)
    c.draw(win)

    win.getMouse() # Pause to view result
    win.close()    # Close window when done

main()
```

width *height*

```
from graphics import *

def main():
    win = GraphWin("CSC111", 600, 400)
    c = Circle(Point(50,50), 10)
    c.draw(win)

    win.getMouse() # Pause to view result
    win.close()    # Close window when done

main()
```

Objects to Play With:

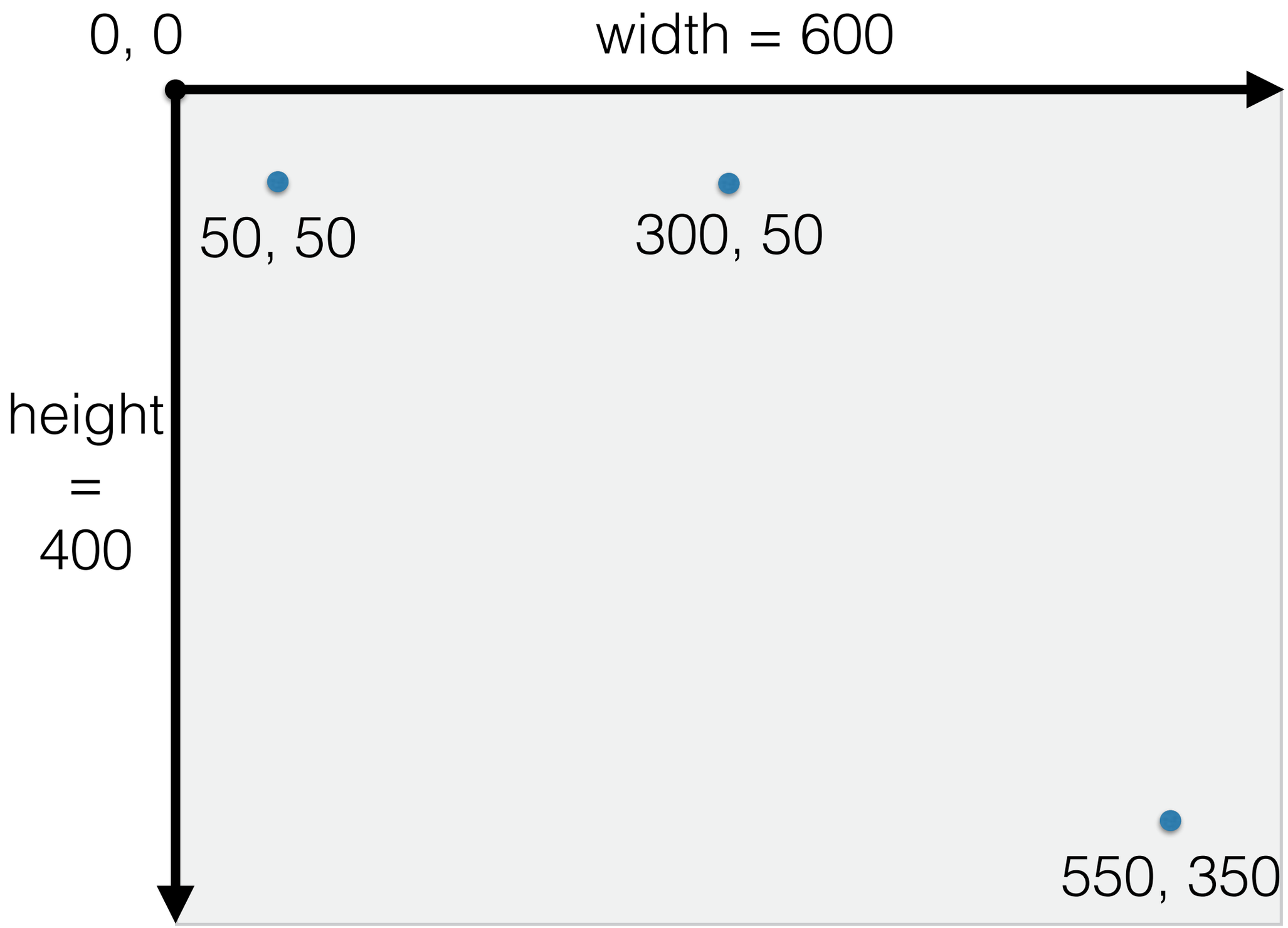
- Points
- Circles
- Rectangles
- Labels (text)

Points

- Used to anchor other objects (circles or rectangles)
- Defined by **x** and **y** coordinates

```
# create a point at location (50, 50)
p1 = Point(50,50)

# create a point at location (300, 50)
p2 = Point(300,50)
```

We stopped here last time…


0, 0
width = 600
50, 50
300, 50
height
=
400
550, 350

Circles

- Defined by a **center** and a **radius**
- The center is a **Point**

```
# create a circle centered at (50, 50)
# with radius 70
c1 = Circle( Point(50,50), 70 )
c1.draw( win )
```

Rectangles

- Defined by a **top-left,** and a **bottom-right point**

```
# create a rectangle with top-left corner
# at (5,5) and bottom-right at (50,50)

r3 = Rectangle( Point(5,5), Point( 50, 50) )
r3.draw( win )
```

Labels

- Defined by an anchor **Point**, and
- A **string** that is displayed, centered on the anchor point.

```
# Create a text label centered at (100,100)
# and containing "Smith College"

label3 = Text( Point(100,100), "Smith College" )
label3.draw( win )
```

Filling an Object with Color

```
# create a rectangle with top-left corner
# at (5,5) and bottom-right at (50,50)

r3 = Rectangle( Point(5,5), Point( 50, 50) )
r3.setFill( "red" )
r3.draw( win )
```

Lot's of Colors to Choose from

http://cs.smith.edu/dftwiki/index.php/Tk_Color_Names

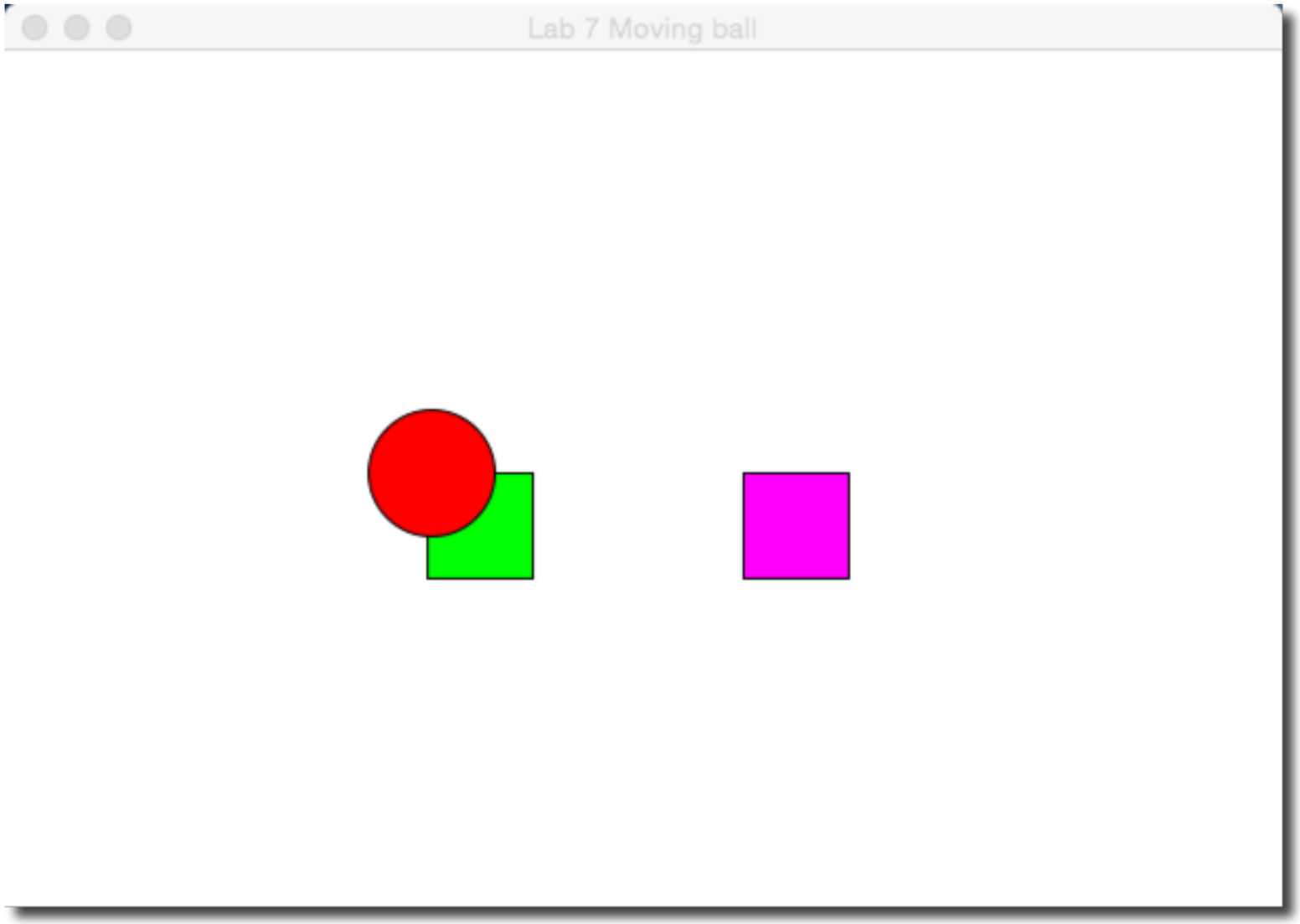
Filling an Object with an RGB Color

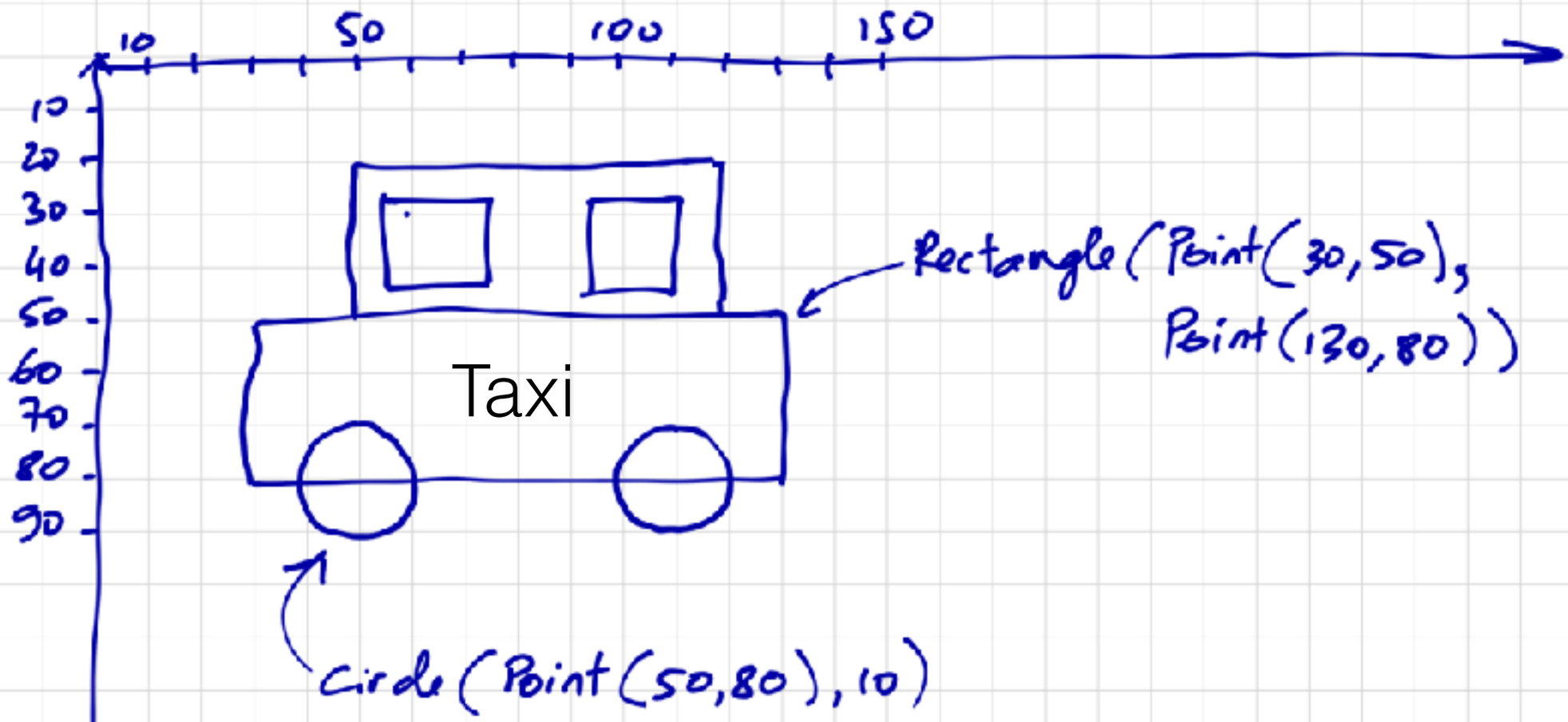
```
# create a rectangle with top-left corner
# at (5,5) and bottom-right at (50,50)

r3 = Rectangle( Point(5,5), Point( 50, 50) )
color = color_rgb( 200, 100, 150 )
r3.setFill( color )
r3.draw( win )
```

Random Colors

```python
# randomColor1.py
# your name here
# generates a rectangle with a random width
# and height, and a given color on the screen.

from graphics import *
from random import *

def main():
    win = GraphWin("Lab 7", 600,600)

    # create rectangle with these 2 corners
    r = Rectangle( Point(50,50), Point(300,300) )

    # create a color from 3 different RGB values
    red   = randint( 0, 255 )
    green = randint( 0, 255 )
    blue  = randint( 0, 255 )
    color = color_rgb( red, green, blue )

    # set the rectangle's color with this color
    r.setFill( color )

    # draw the rectangle
    r.draw( win )

    # wait for user to click on the window before closing
    win.getMouse()
    win.close()

main()
```

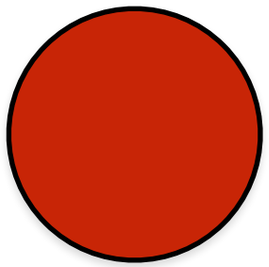
Demo Time

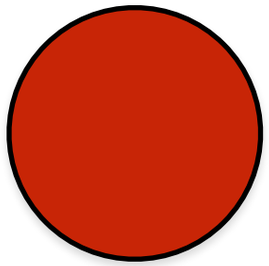


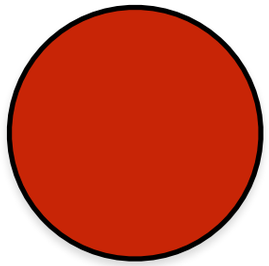


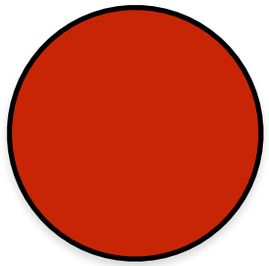
Drawing a Taxi Cab (Lab 7)

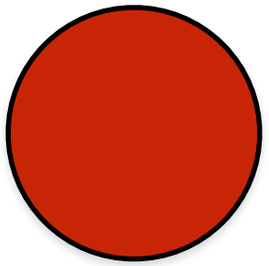
Basics of Animation

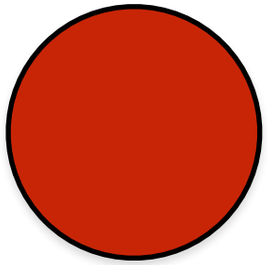




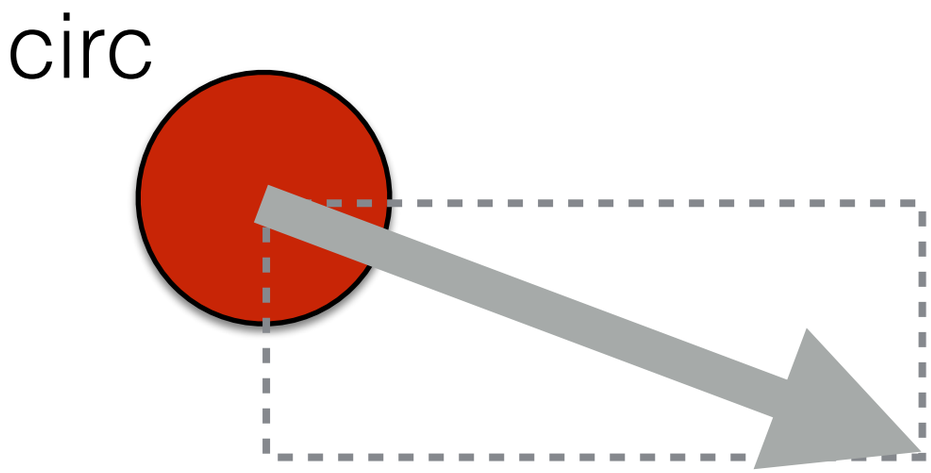




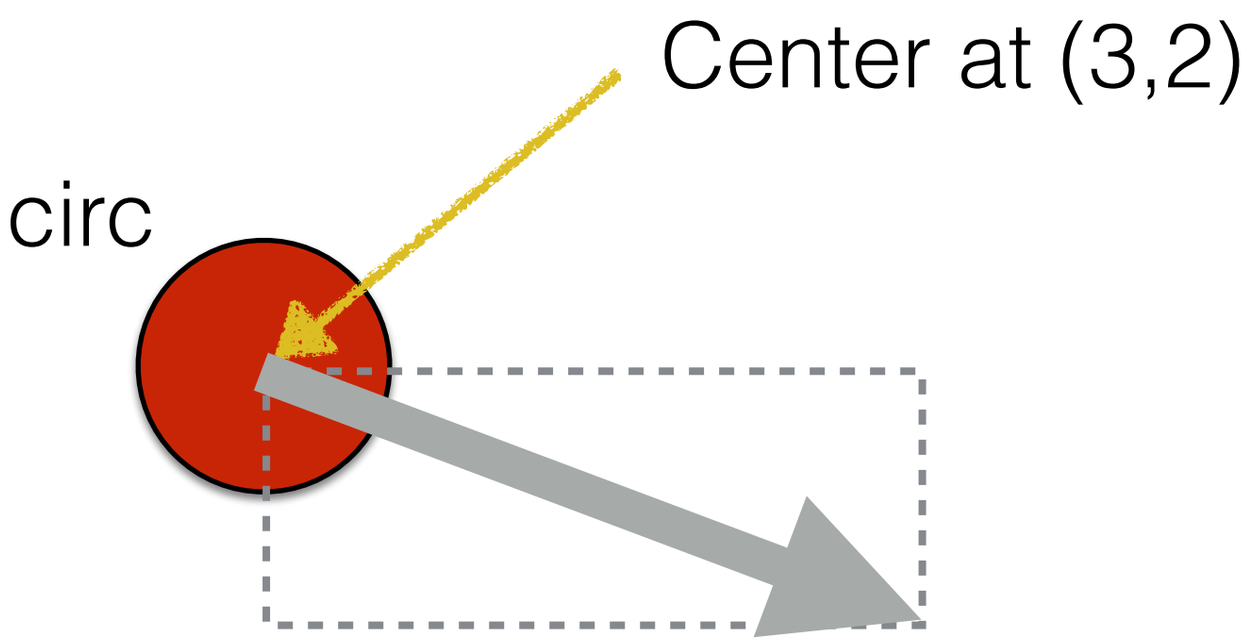




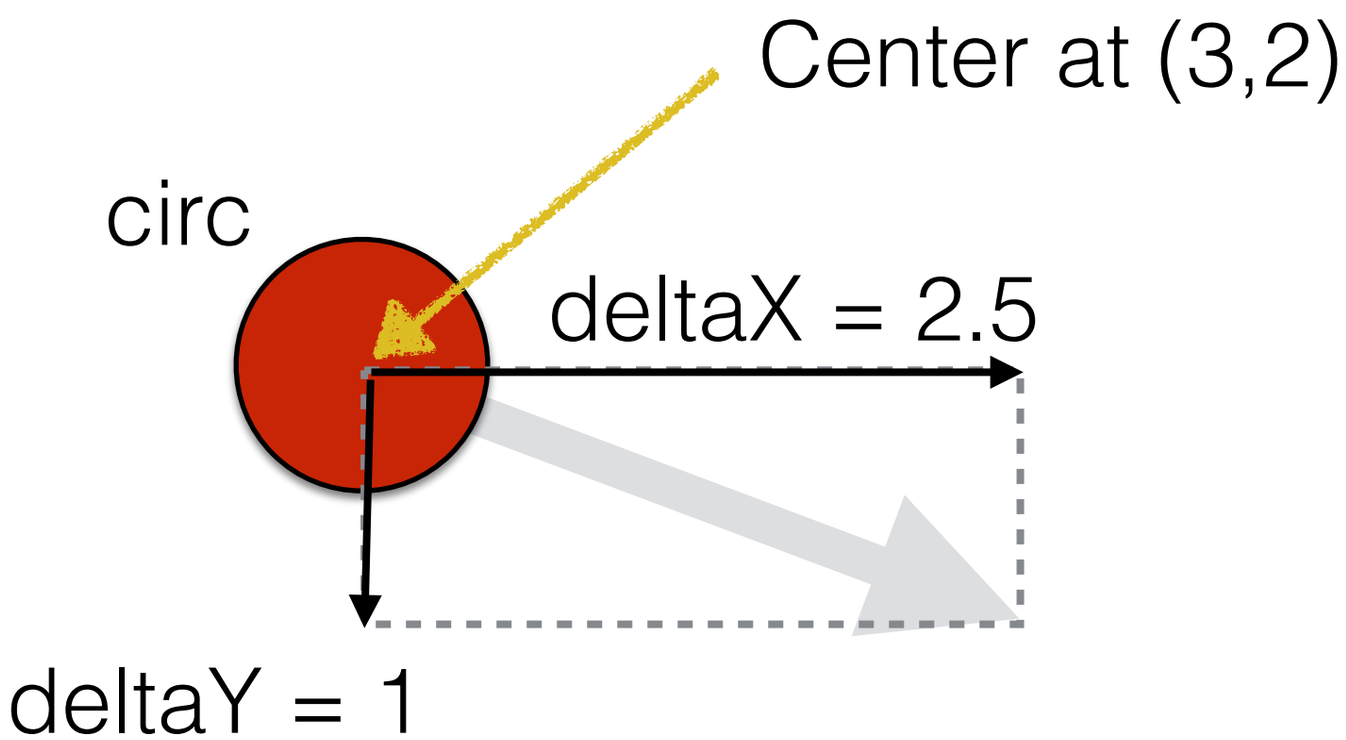
Understanding Motion



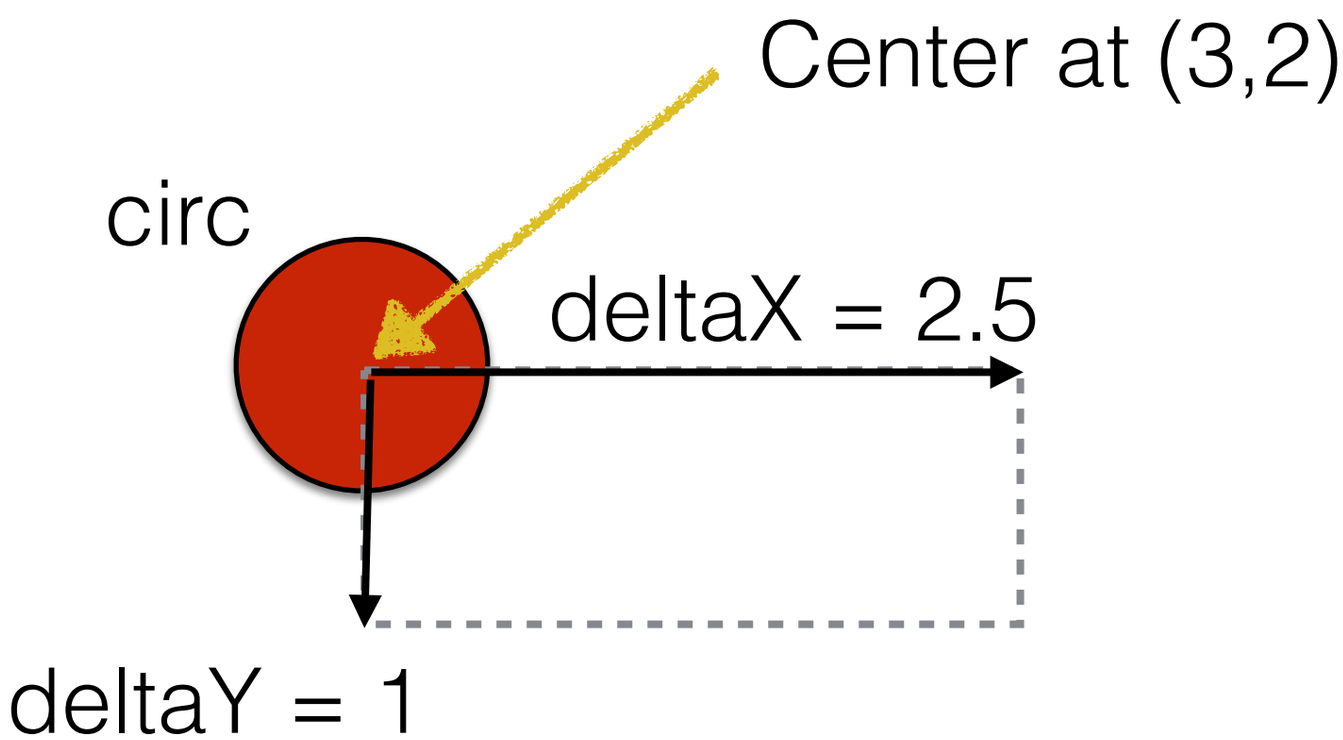
Understanding Motion



Understanding Motion

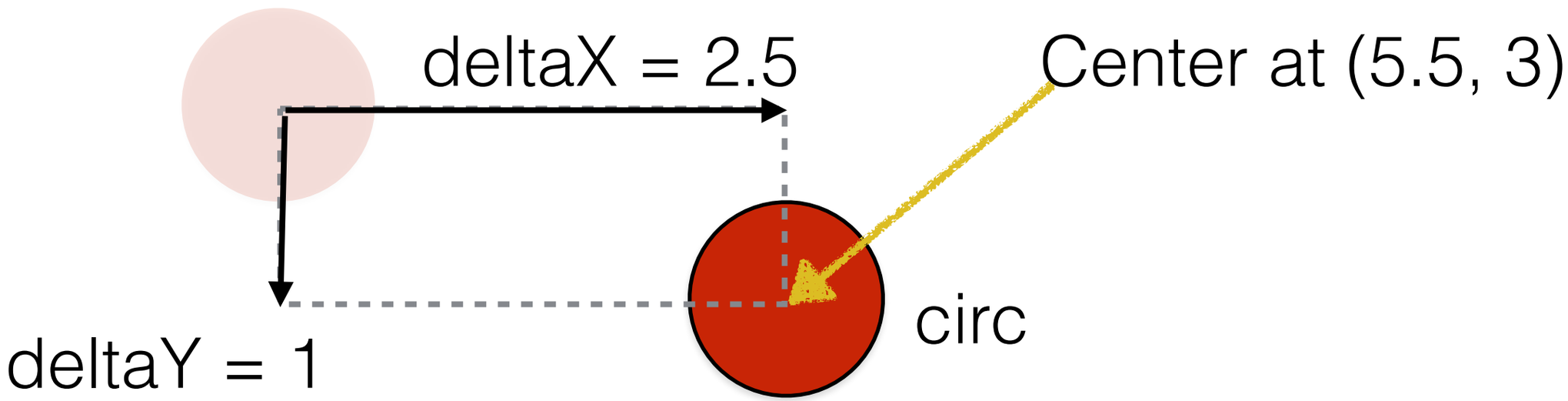


Understanding Motion



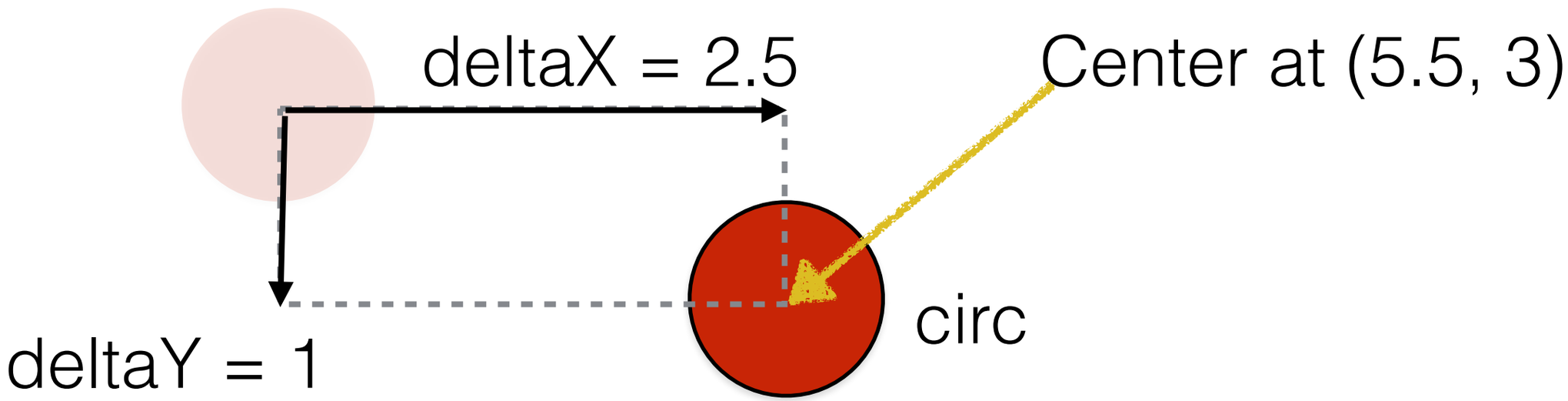

```
circ.move( deltaX, deltaY )
```

Understanding Motion



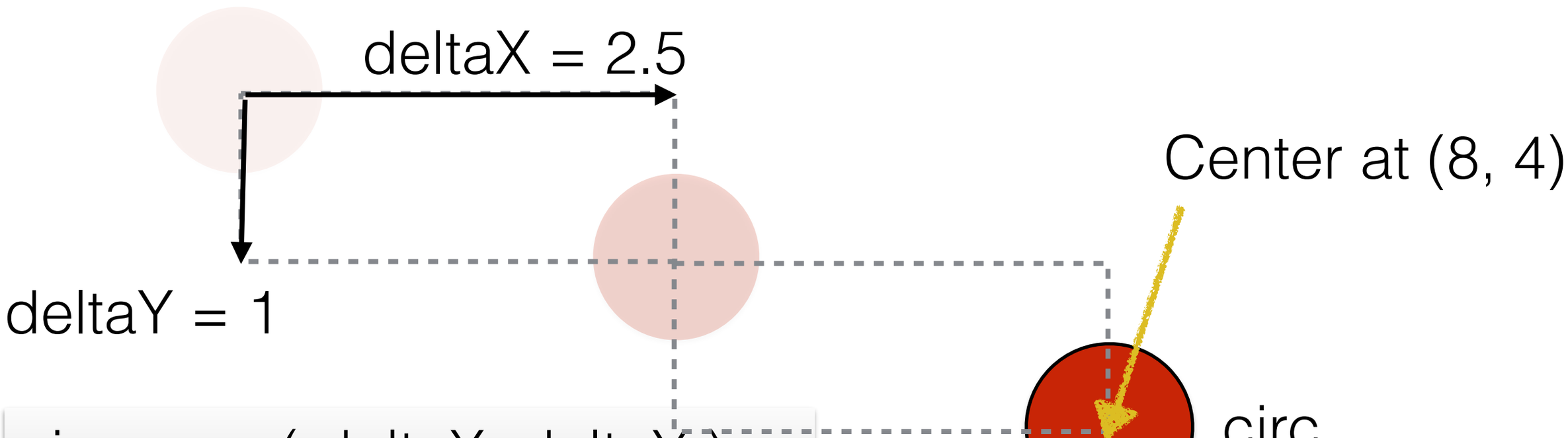

```
circ.move( deltaX, deltaY )
```

Understanding Motion




```
circ.move( deltaX, deltaY )
circ.move( deltaX, deltaY )
```

Understanding Motion




```
circ.move( deltaX, deltaY )
circ.move( deltaX, deltaY )
```

```python
from graphics import *

def main():
    win = GraphWin( "Lab 7 Moving ball", 600, 400 )

    # create and draw a red circle
    center = Point( 100, 100 )
    circ     = Circle( center, 30 )
    circ.setFill( 'red' )
    circ.draw( win )

    # set initial direction of ball
    dx = 1.5
    dy = 0.25

    # move ball on screen
    while win.checkMouse() == None:
        circ.move( dx, dy )
        #x = circ.getCenter().getX()
        #y = circ.getCenter().getY()

    win.close()    # Close window when done

main()
```

How can we test that an object is moving out of the graphics window?

Every graphic element is an OBJECT

Examples

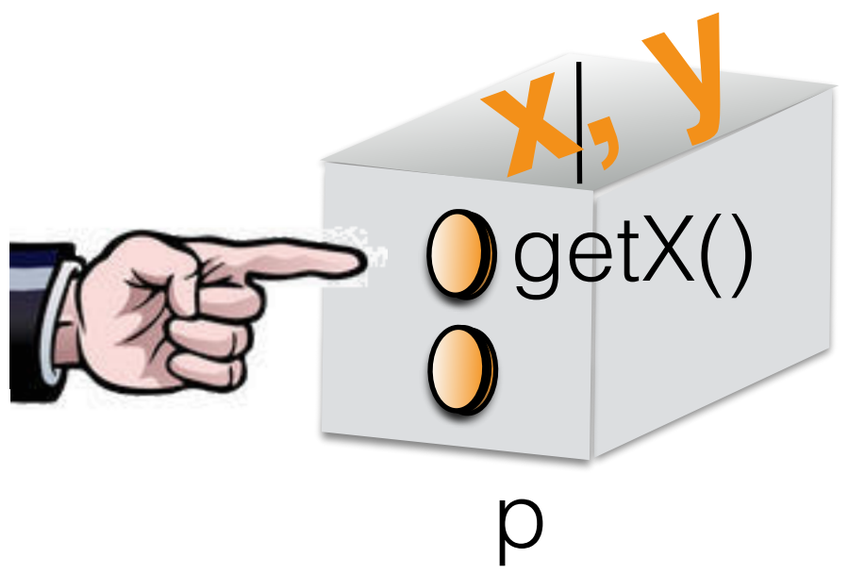
Organization of a graphic program

Something completely different...

Every Graphic Element *is* an Object

Point

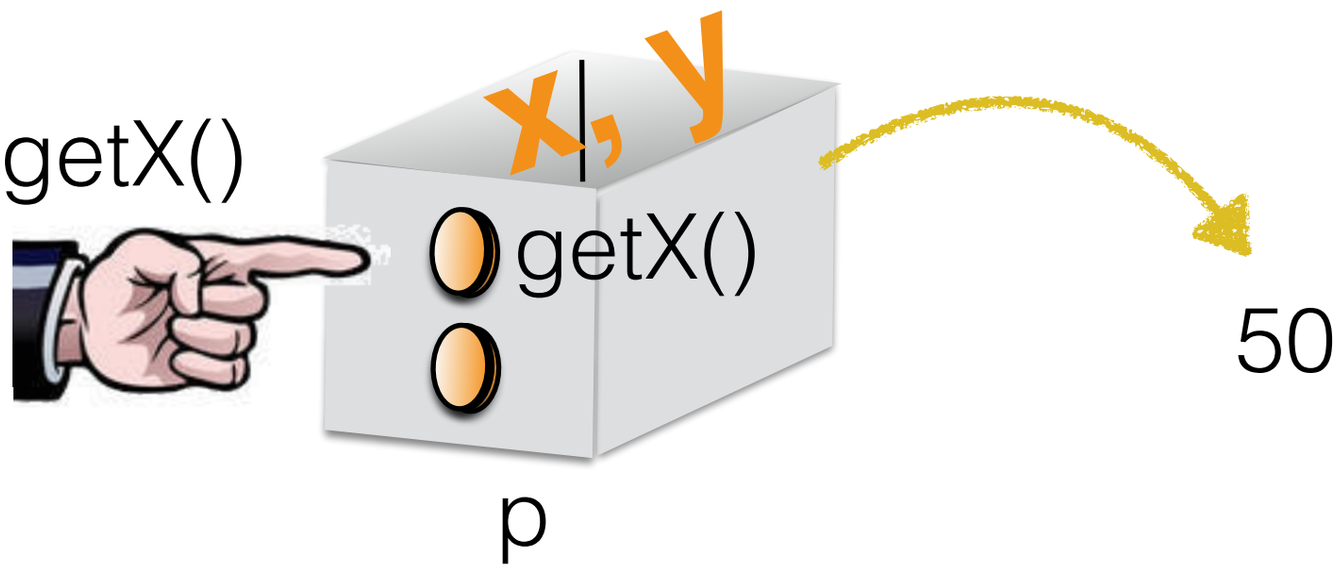
p = Point(50, 150)



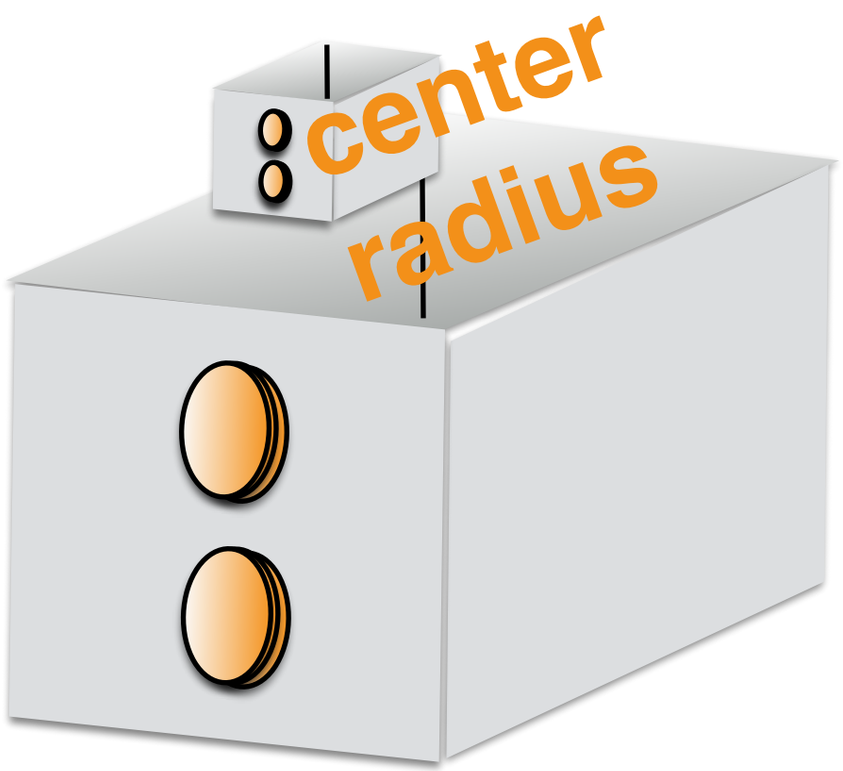
Point

p = Point(50, 150)

p.getX()

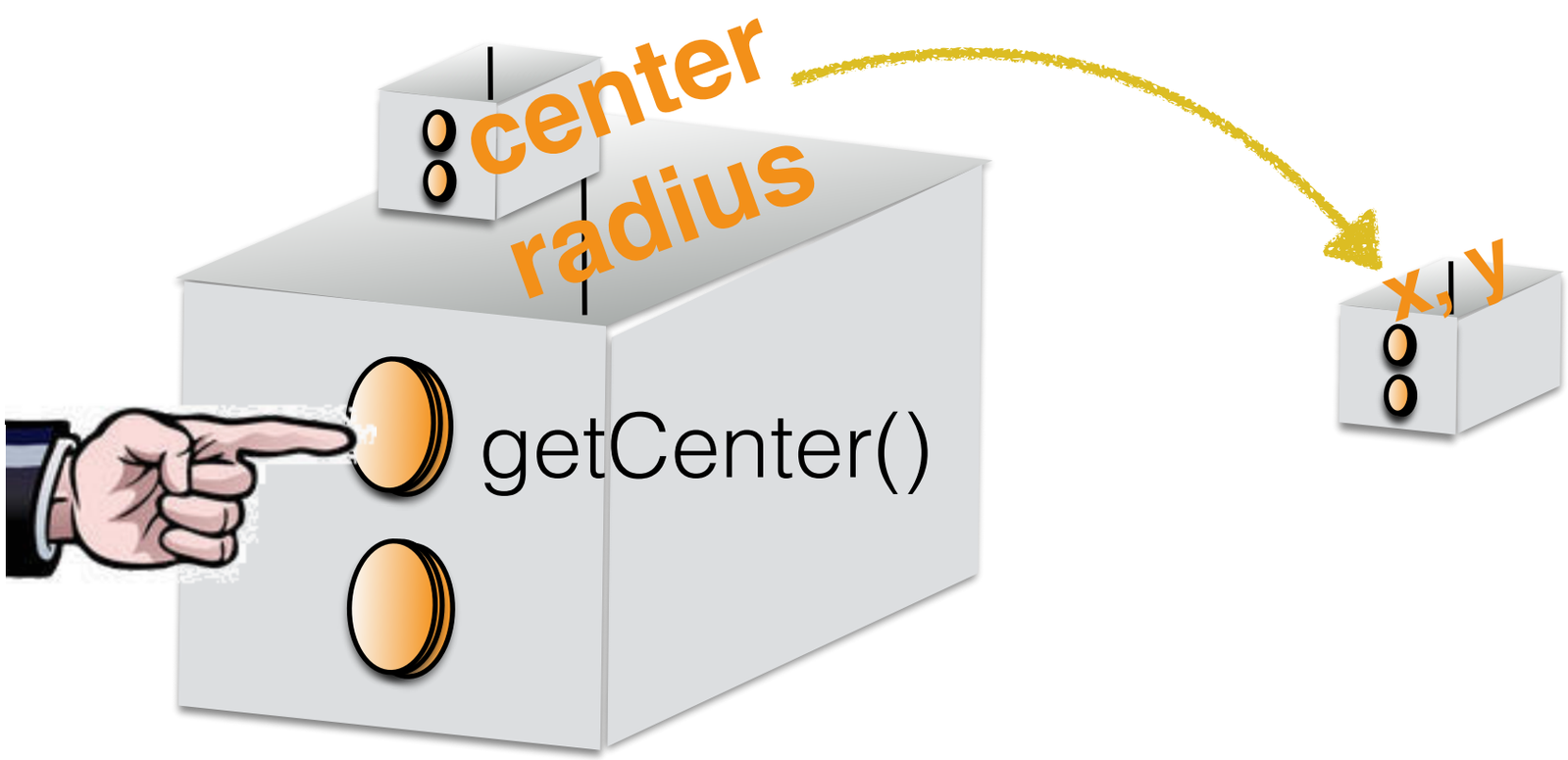


Circle



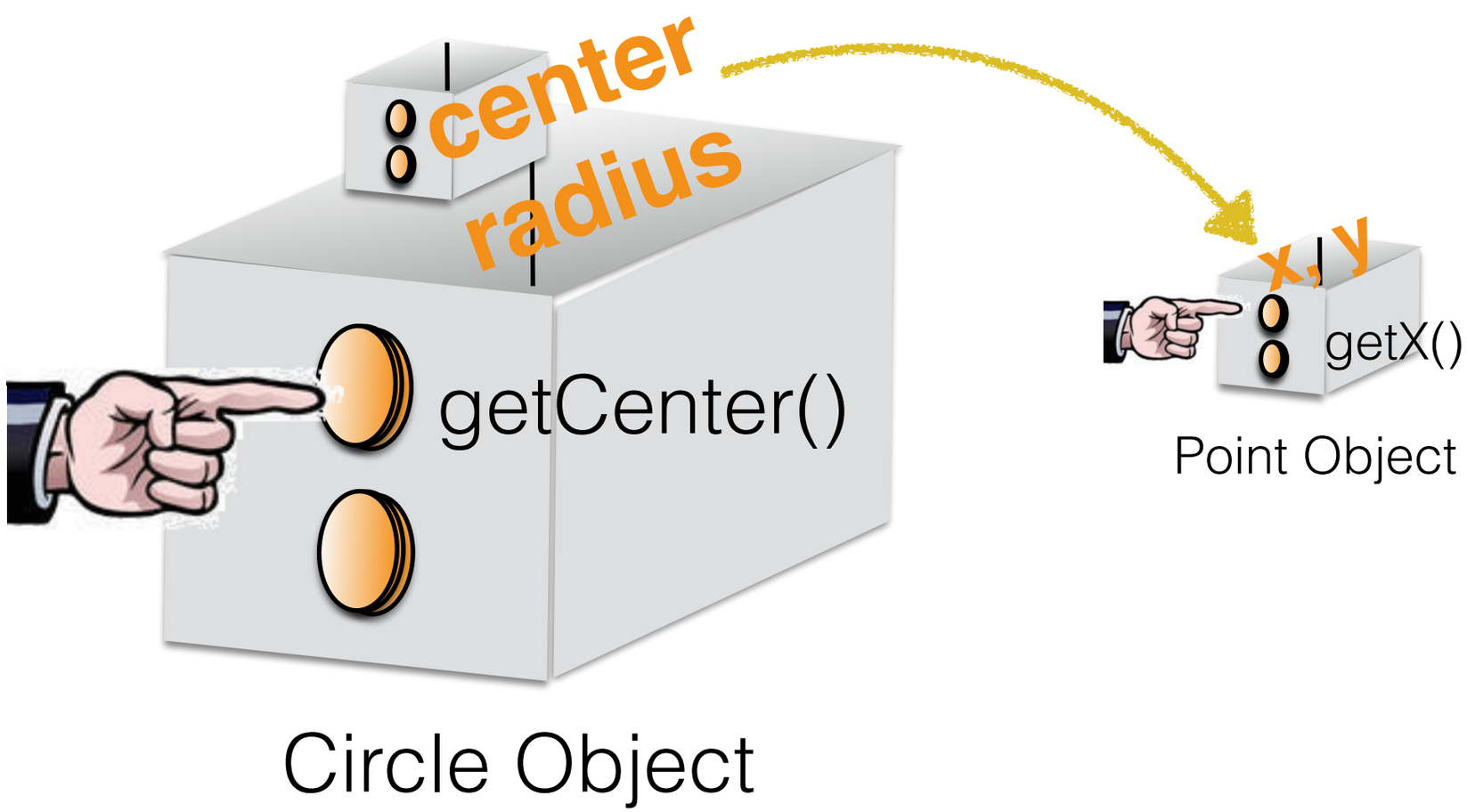
Circle Object

Circle

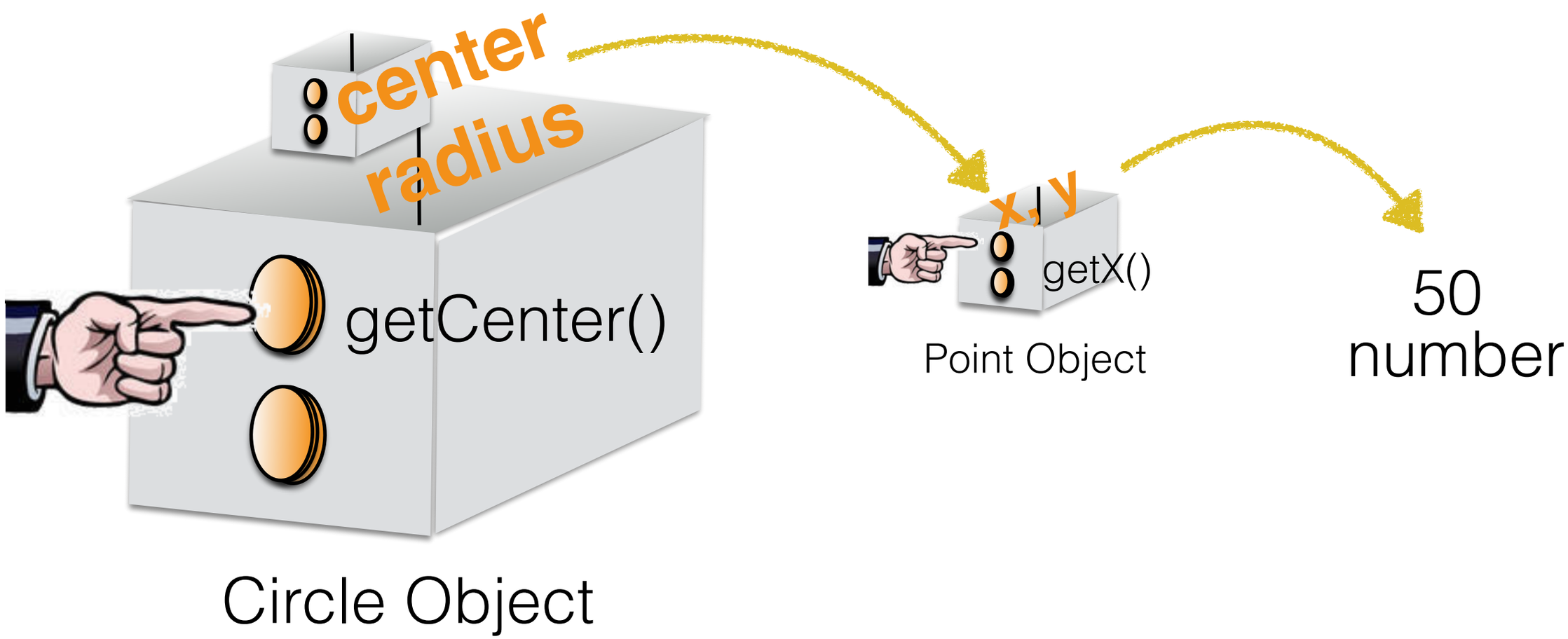


Circle Object

Circle

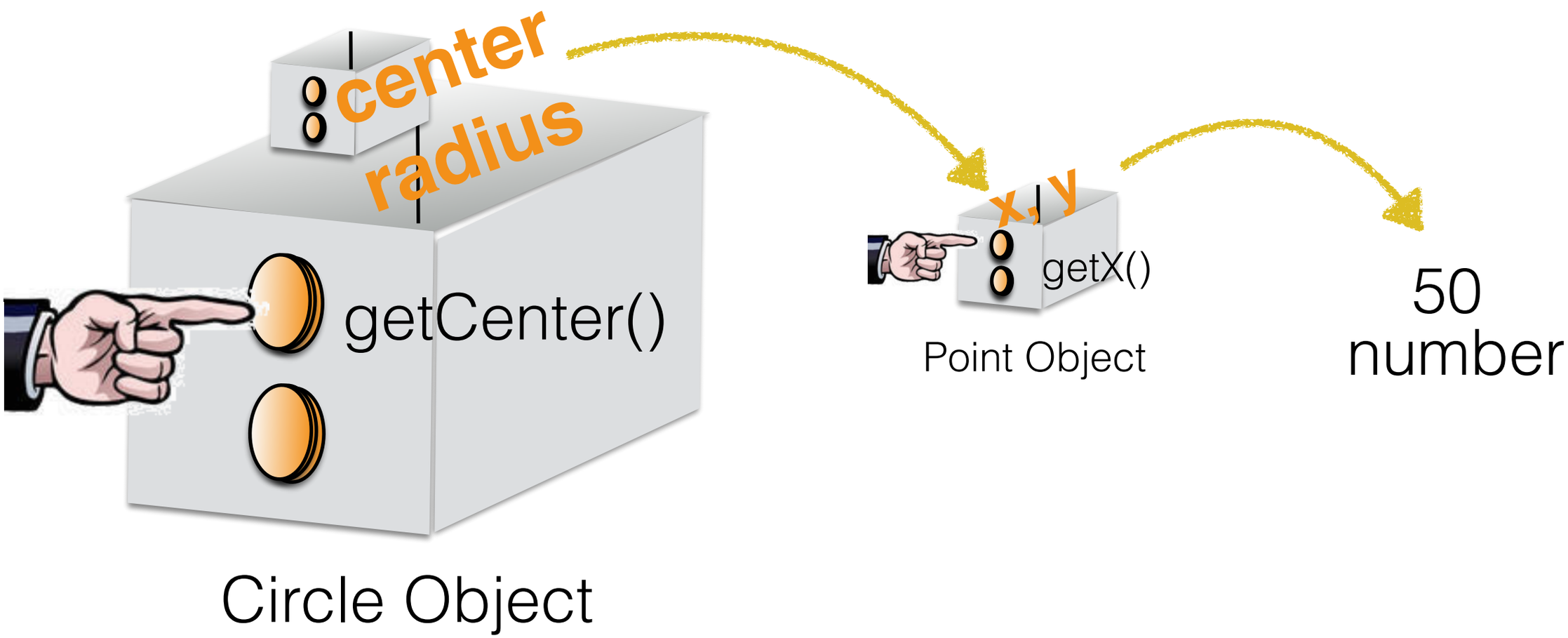

center
radius
getCenter()
x,y
getX()
Point Object
Circle Object

Circle

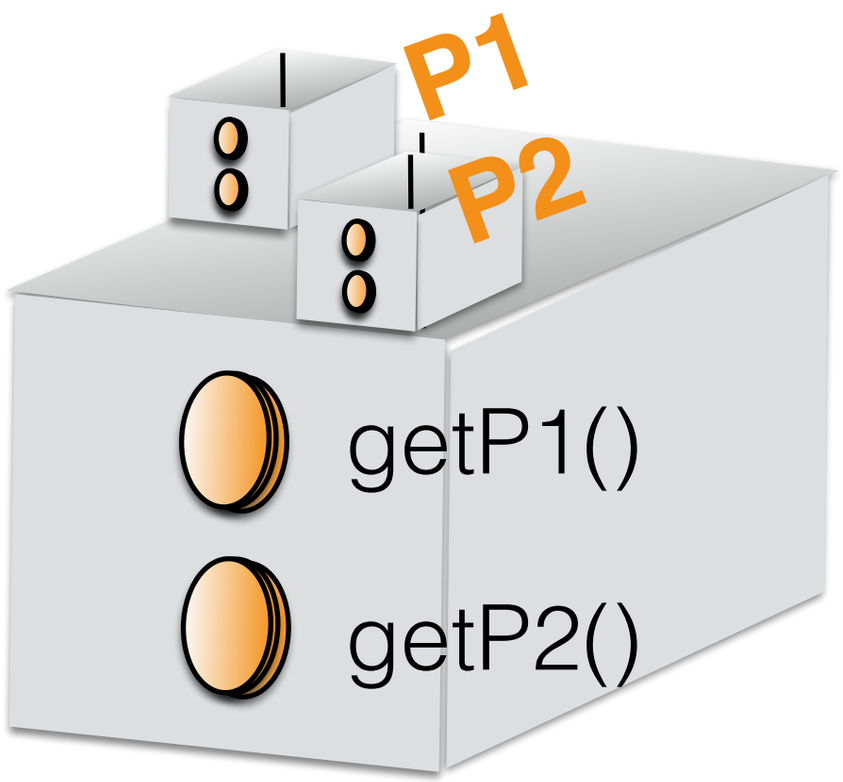

center
radius
getCenter()
Circle Object
x,y
getX()
Point Object
50
number

Circle

```
x = circ.getCenter().getX()
```

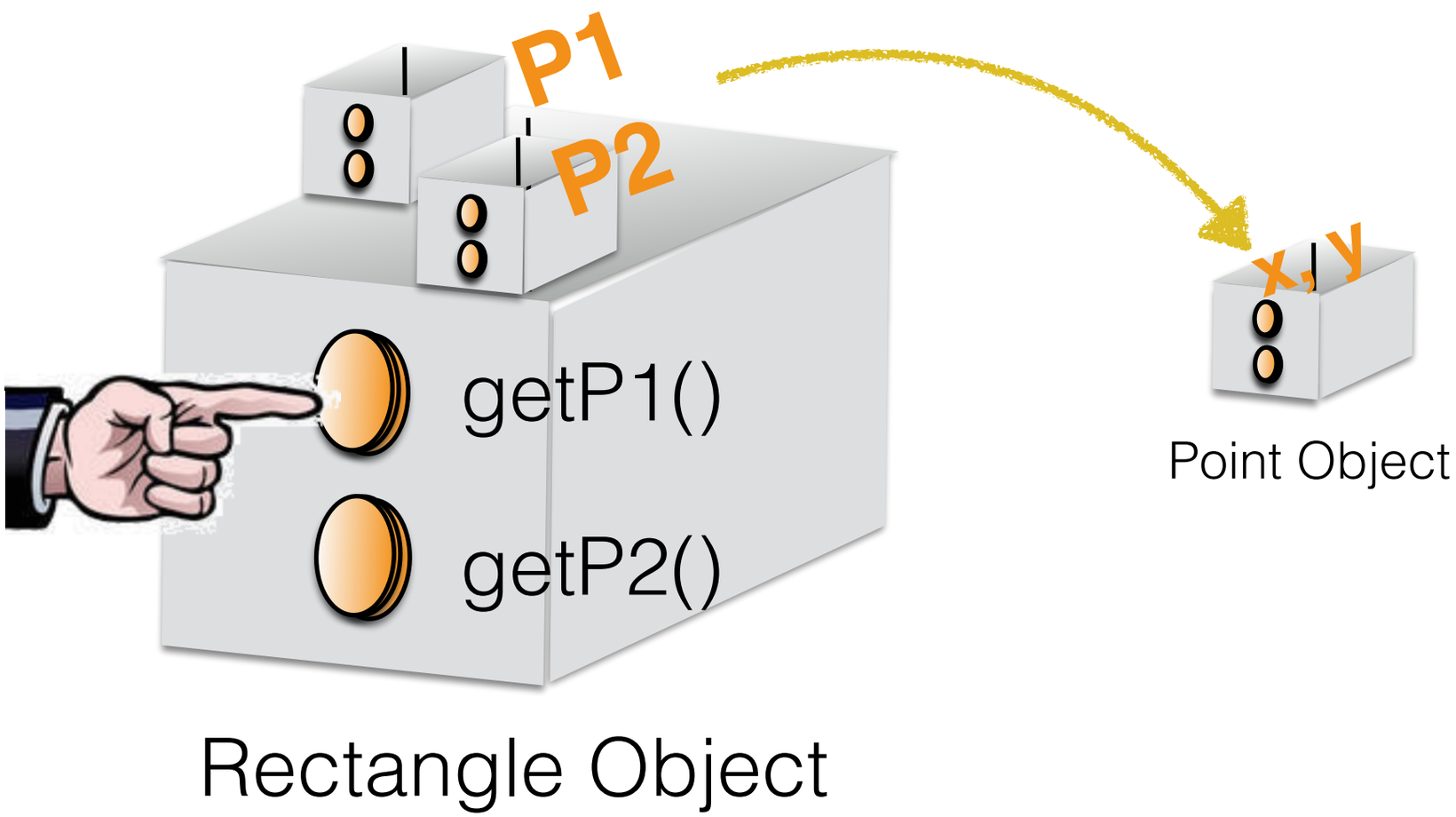


Rectangle

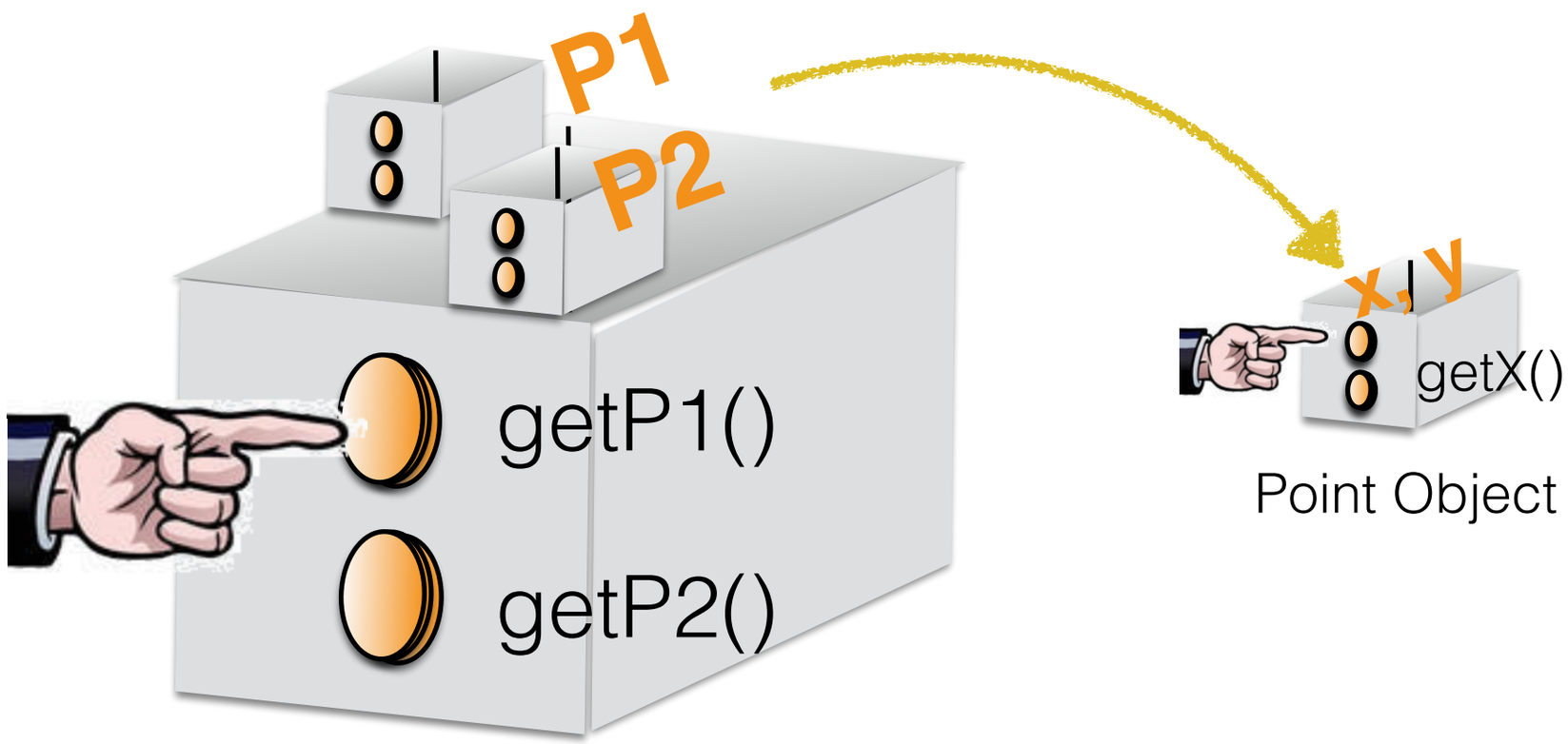


Rectangle Object

Rectangle

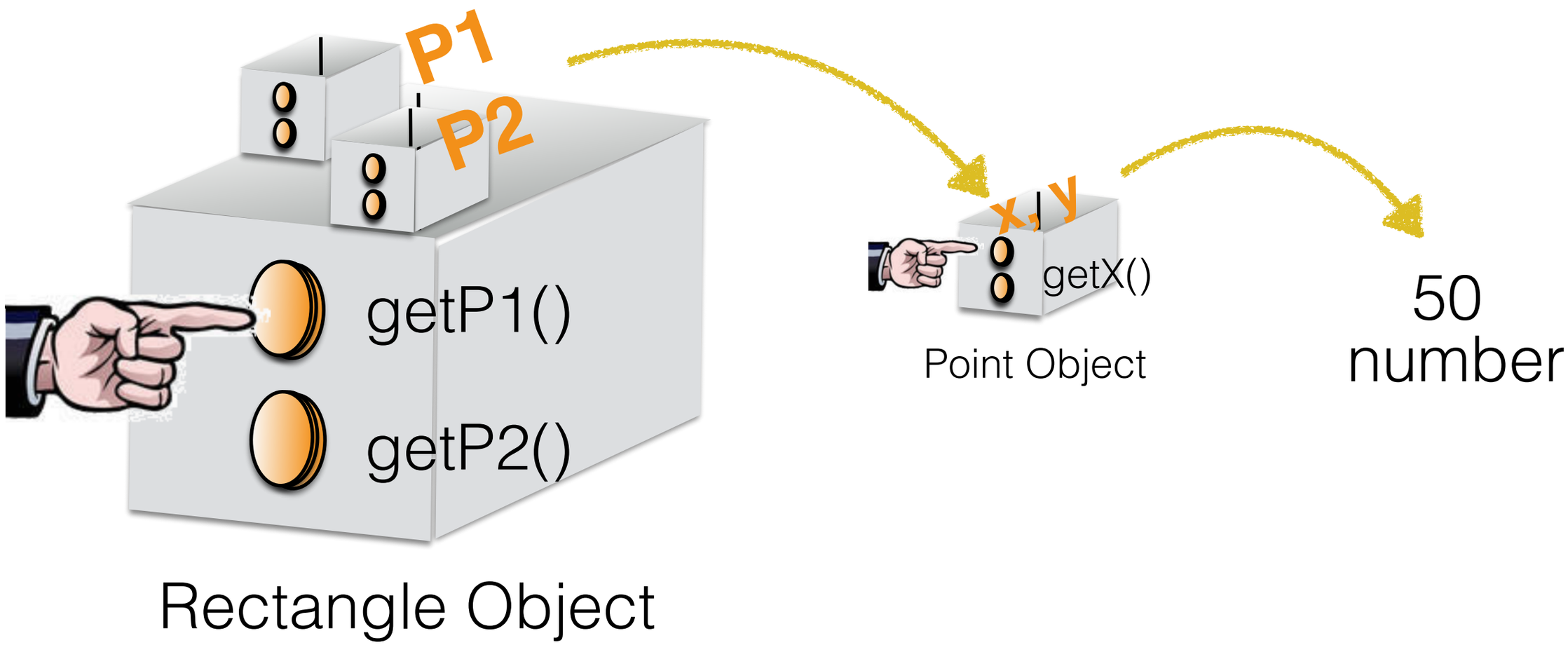


Rectangle

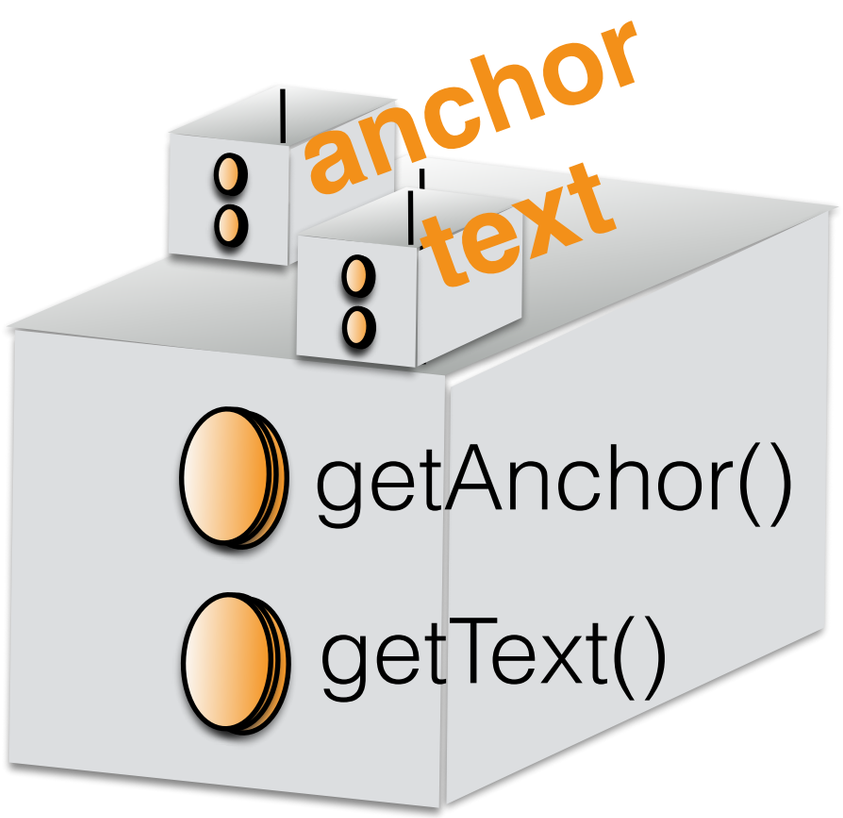


Rectangle Object

Rectangle

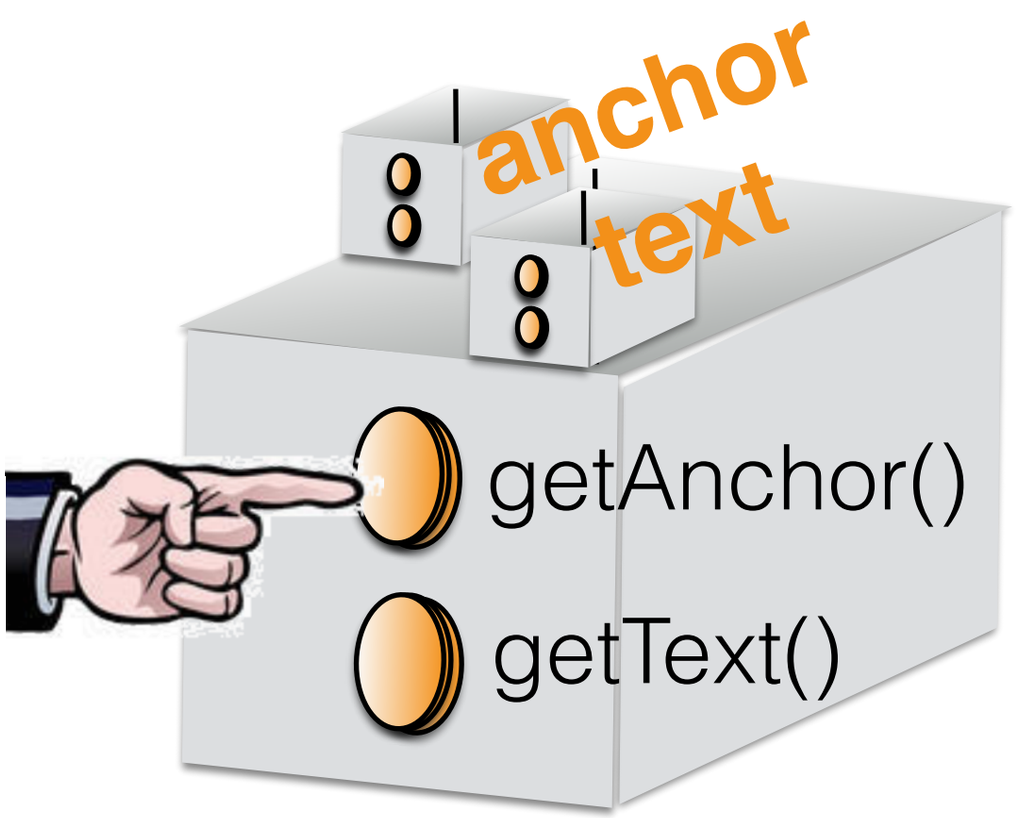

P1
P2
getP1()
getP2()
Rectangle Object
x, y
getX()
Point Object
50
number

Text (label)



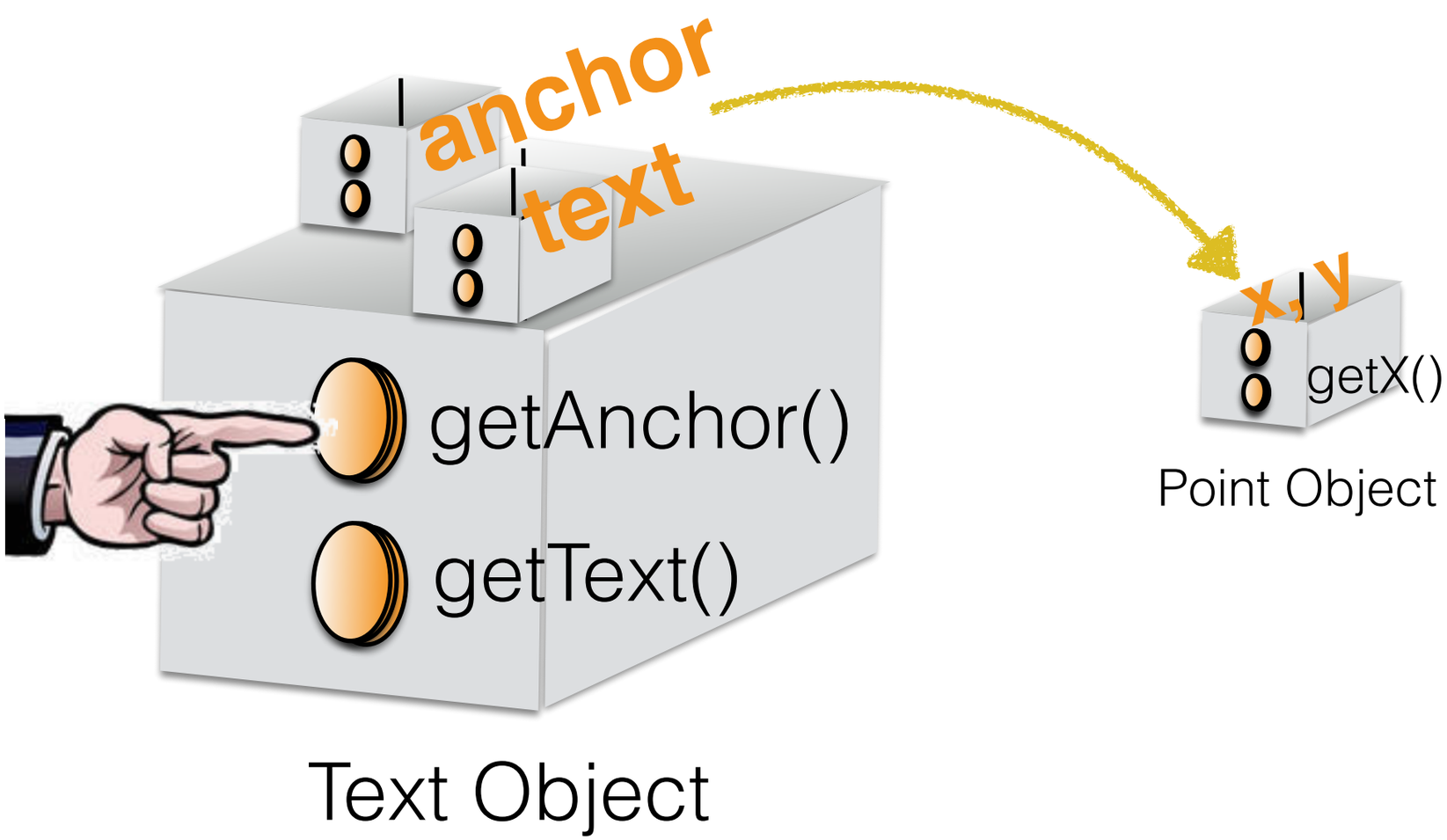
Text Object

Text (label)

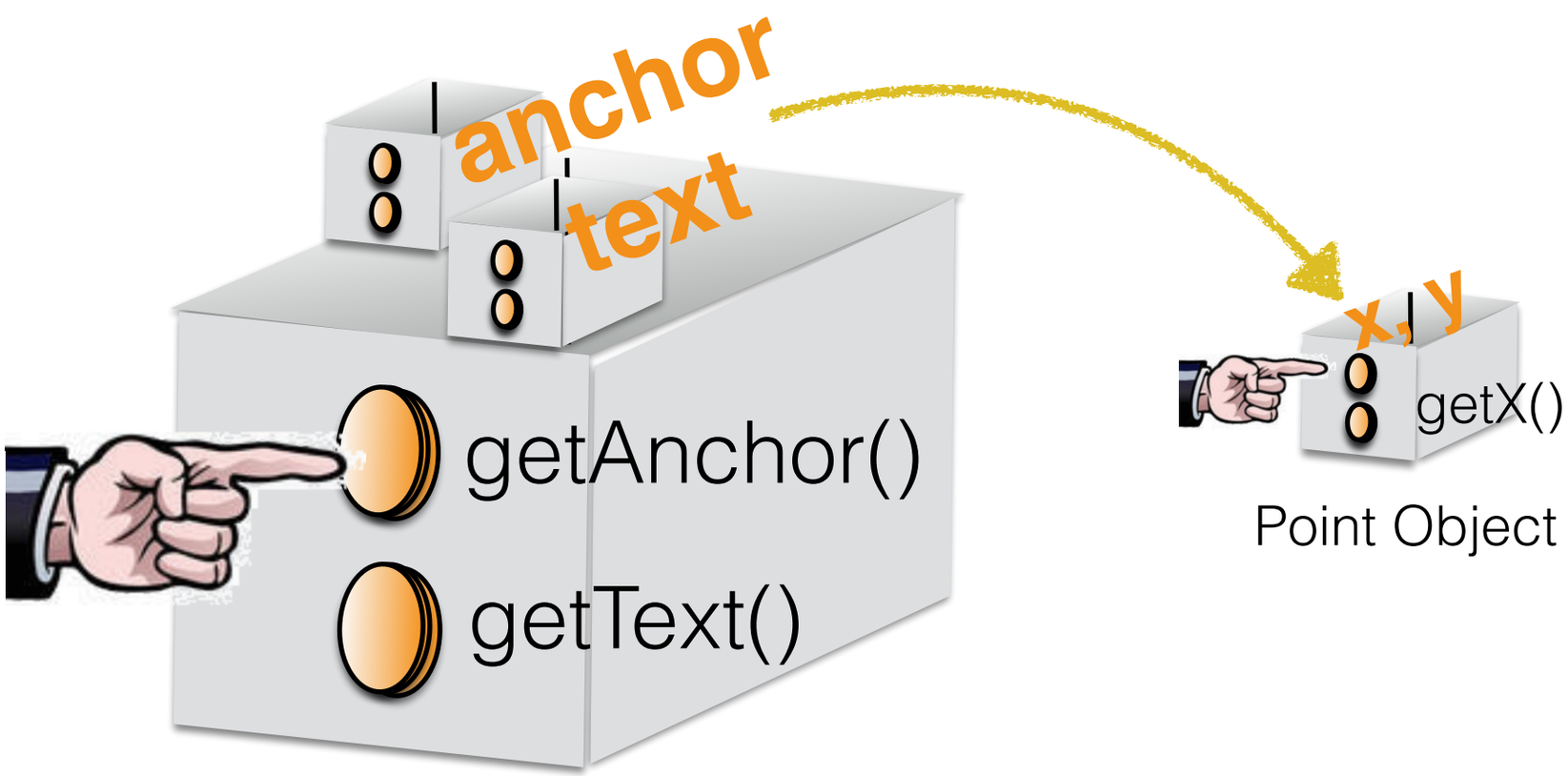


Text Object

Text (label)


anchor
text
getAnchor()
getText()
Text Object
x, y
getX()
Point Object

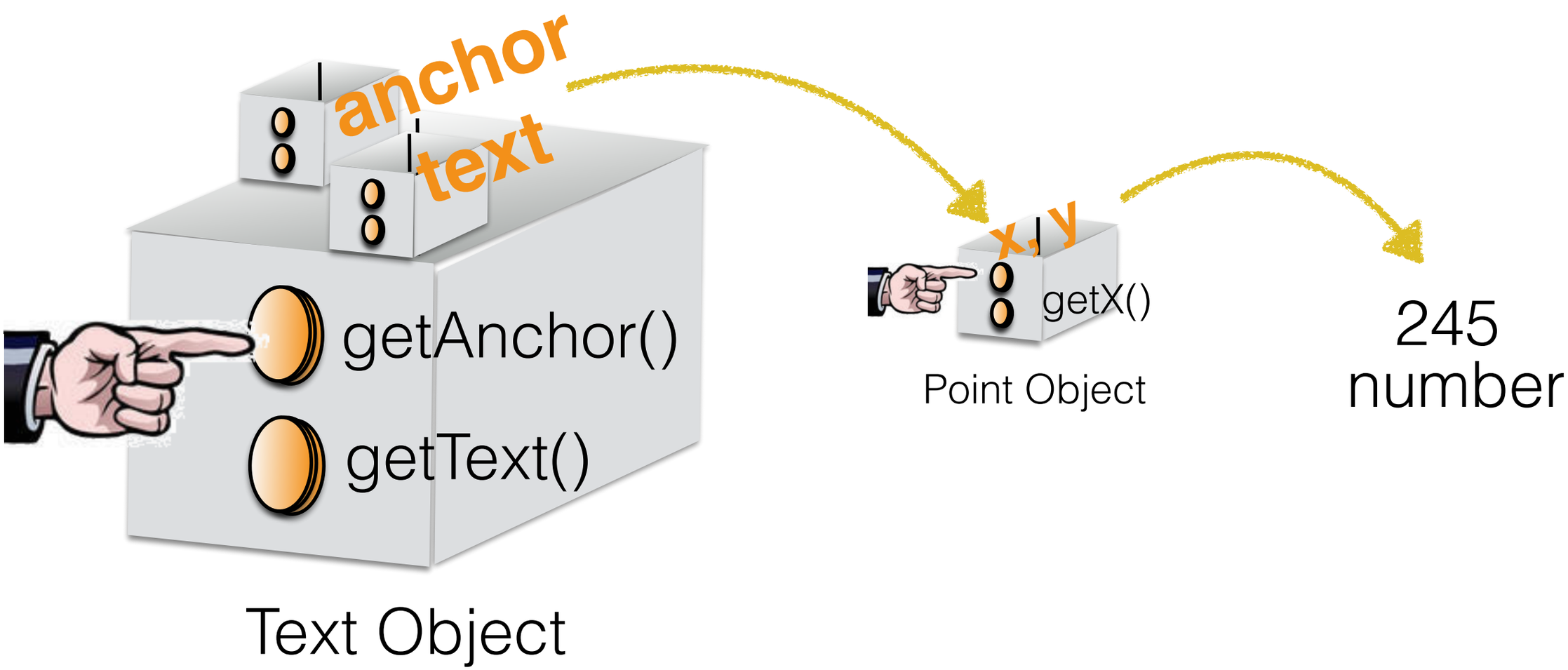
Text (label)



Text Object

Point Object

Text (label)


anchor
text
getAnchor()
getText()
Text Object
x,y
getX()
Point Object
245
number

We stopped here last time…

Take the Survey!

—>Emma Stephenson

—>Piazza

Click Here!

Diffchecker Home Images PDF Contact CLI Desktop Sign in Create a free accou

RECENT DIFFS

Unsaved diff

Clear diffs

Recent diffs are deleted on refresh

SAVED DIFFS

You must log in to save diffs

9 removals 10 additions

```
 1. Who should we sing for?
 2. +---------------------------+
 3. |   Happy birthday to you!  |
 4. |   Happy birthday to you!  |
 5. |Happy birthday dear, Murphy!|
 6. |   Happy birthday to you!  |
 7. +---------------------------+
 8. +---------------------------+
 9. |   Happy birthday to you!  |
10. |   Happy birthday to you!  |
11. |Happy birthday dear, Maria!|
12. |   Happy birthday to you!  |
13. +---------------------------+
14. +---------------------------+
15. |   Happy birthday to you!  |
16. |   Happy birthday to you!  |
17. |Happy birthday dear, Allie!|
18. |   Happy birthday to you!  |
19. +---------------------------+
20.
```

```
 1. Who should we sing for?
 2. +---------------------------+
 3. |   Happy birthday to you!  |
 4. |   Happy birthday to you!  |
 5. |Happy birthday, dear Murphy!|
 6. |   Happy birthday to you!  |
 7. +---------------------------+
 8. +---------------------------+
 9. |  Happy birthday to you!   |
10. |  Happy birthday to you!   |
11. |Happy birthday, dear Maria!|
12. |  Happy birthday to you!   |
13. +---------------------------+
14. +---------------------------+
15. |  Happy birthday to you!   |
16. |  Happy birthday to you!   |
17. |Happy birthday, dear Allie!|
18. |  Happy birthday to you!   |
19. +---------------------------+
20.
```

ORIGINAL TEXT

```
 1 Who should we sing for?
 2 +---------------------------+
 3 |   Happy birthday to you!  |
 4 |   Happy birthday to you!  |
 5 |Happy birthday dear, Murphy!|
 6 |   Happy birthday to you!  |
 7 +---------------------------+
 8 +---------------------------+
 9 |   Happy birthday to you!  |
10 |   Happy birthday to you!  |
11 |Happy birthday dear, Maria!|
12 |   Happy birthday to you!  |
13 +---------------------------+
14 +---------------------------+
15 |   Happy birthday to you!  |
16 |   Happy birthday to you!  |
17 |Happy birthday dear, Allie!|
18 |   Happy birthday to you!  |
19 +---------------------------+
20
```

CHANGED TEXT

```
 1 Who should we sing for?
 2 +---------------------------+
 3 |   Happy birthday to you!  |
 4 |   Happy birthday to you!  |
 5 |Happy birthday, dear Murphy!|
 6 |   Happy birthday to you!  |
 7 +---------------------------+
 8 +---------------------------+
 9 |  Happy birthday to you!   |
10 |  Happy birthday to you!   |
11 |Happy birthday, dear Maria!|
12 |  Happy birthday to you!   |
13 +---------------------------+
14 +---------------------------+
15 |  Happy birthday to you!   |
16 |  Happy birthday to you!   |
17 |Happy birthday, dear Allie!|
18 |  Happy birthday to you!   |
19 +---------------------------+
20
21
```

If Statements

Chapter 7 in Zelle

Bits, binary switch

Relational Operators

Boolean Operators

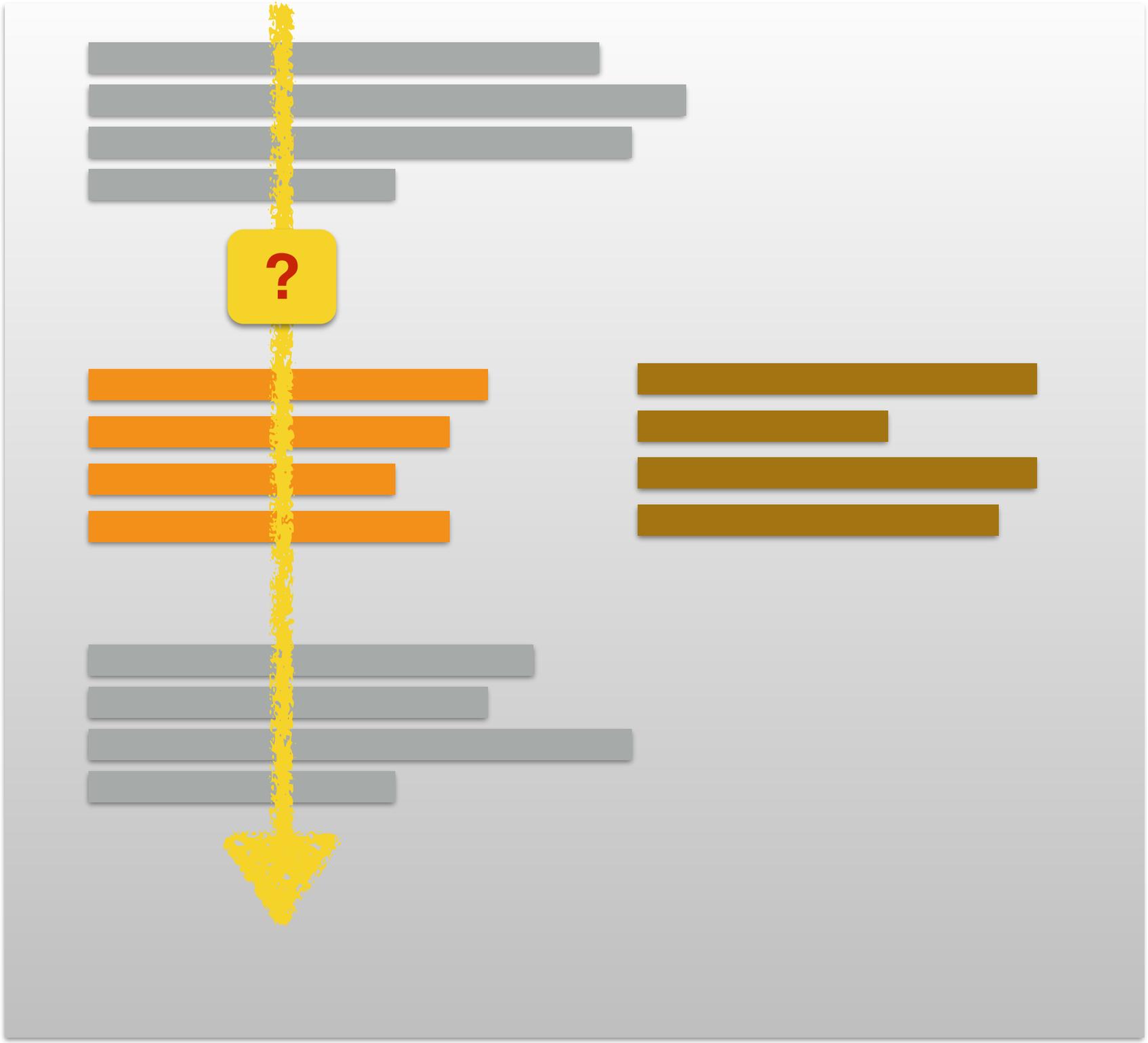
Teller Machine Revisited

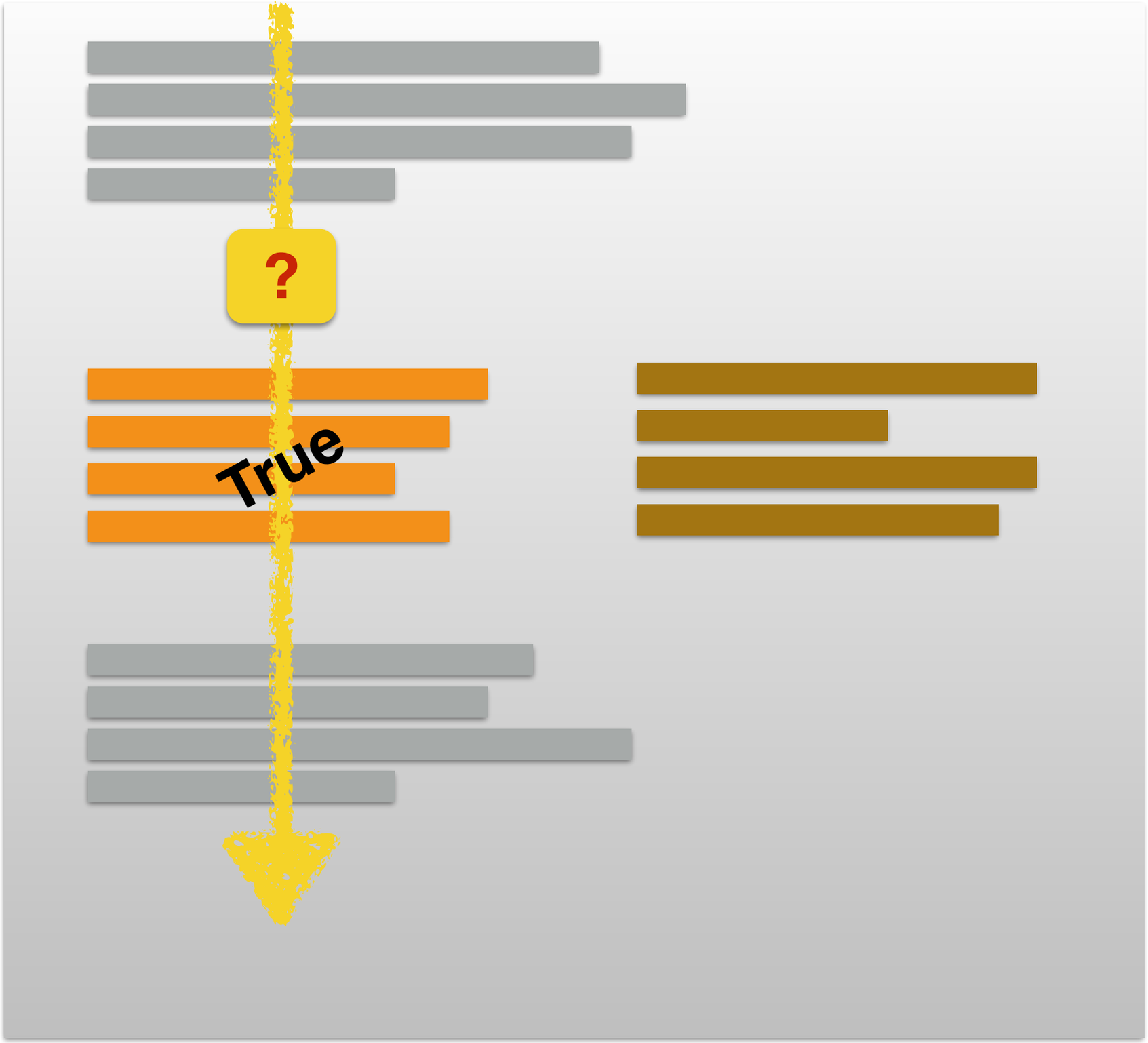
Rock, Paper, Scissors

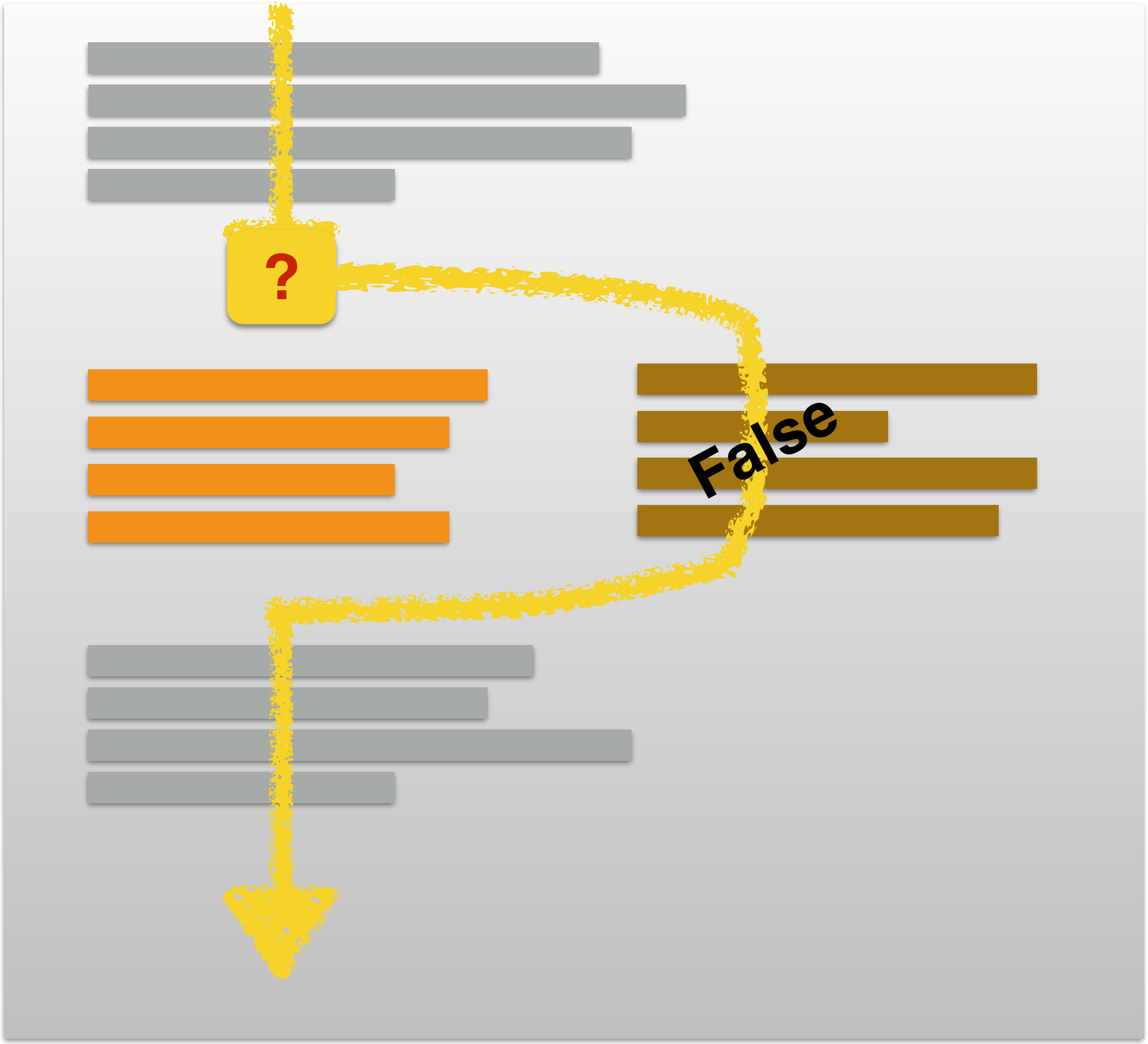
Exercises

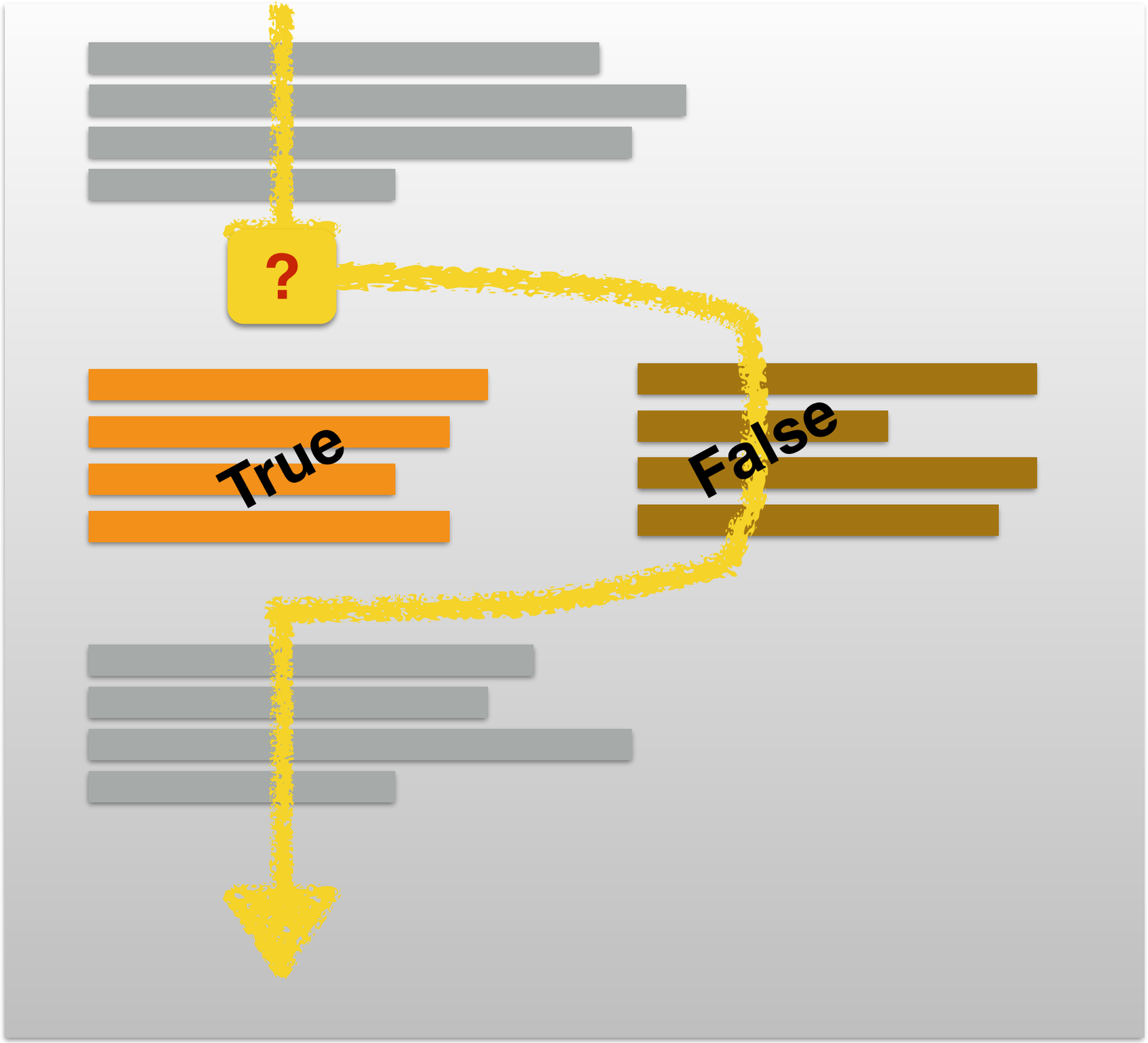
Bits & Boolean Values

- Bits: **0** and **1**
- Boolean values: **True** and **False**
- Boolean switches: Imagine a world where every decision has a binary outcome:
    - Do you want to go out or do you want to stay in?
    - If you go out, do you walk or do you take the car?




?
True


?
False


?
True
False


?
True
False

```
if  answer to question is True:
    True
else:
    False
```

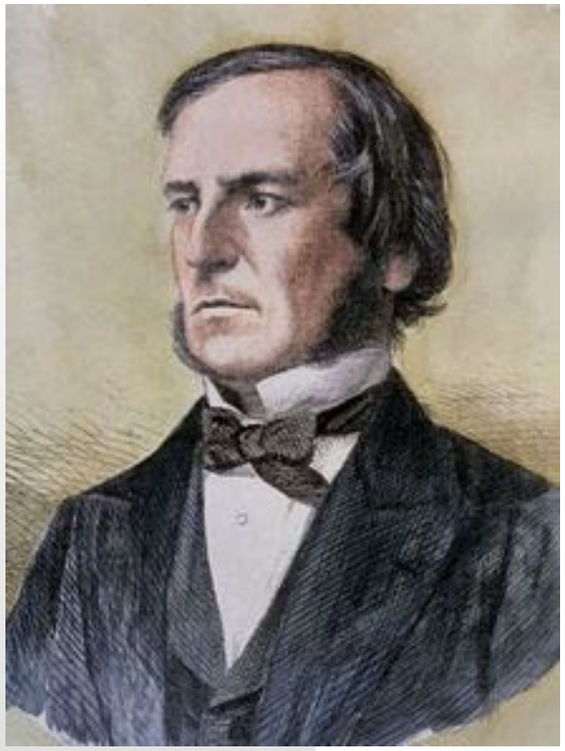

Boolean Expression
if ~~question:~~
True
else:
False

Boolean Expression
if ~~question:~~

True

else:

False



George Boole
1815-1864
(wikipedia.org)

Life Examples

```
if today is school day:
    go to class
else:
    go away for weekend

if food at Duckett > food at your house:
    go eat at Duckett
else:
    go eat at your house

if fire alarm rings:
    evacuate building
```

Life Examples

```
if driving parents' car:
    if low on fuel:
        if close to home:
            drive home and let parents will fill up car
        else:
            stop next gas station and buy minimum gas to get home
```

Life Examples

```
if driving parents' car:
    if low on fuel:
        if close to home:
            drive home and let parents will fill up car
        else:
            stop next gas station and buy minimum gas to get home
```

Life Examples

```
if driving parents' car:
    if low on fuel:
        if close to home:
            drive home and let parents will fill up car
        else:
            stop next gas station and buy minimum gas to get home
```

Python Example

```
circ.move( dx, dy )
x = circ.getCenter().getX()
y = circ.getCenter().getY()
if x > 600:
    dx = -dx
if x < 0:
    dx = -dx
if y > 400:
    dy = -dy
if y < 0:
    dy = -dy
```

Python Exercise

Amount to withdraw? **71**
3 $20-bill(s)
1 $10-bill(s)
0 $5-bill(s)
1 $1-bill(s)

Exercise: print 's' whenever necessary

Python Exercise

```
amount = int( input( "Amount? " ) )

no20s = amount // 20
amount = amount % 20

no10s = amount // 10
amount = amount % 10

no5s = amount // 5
no1s = amount % 5

print( no20s, "$20-bill(s)" )
…
```

Python Exercise

```
amount = int( input( "Amount? " ) )

no20s = amount // 20
amount = amount % 20

no10s = amount // 10
amount = amount % 10

no5s = amount // 5
no1s = amount % 5

if no20s == 1:
   print( no20s, "$20-bill" )
else:
   print( no20s, "$20-bills" )

…
```

Assume no20s contains 1…

True

```
if no20s == 1:
   print( no20s, "$20-bill" )
else:
   print( no20s, "$20-bills" )
```

Assume no20s contains 3…

False

```
if no20s == 1:
    print( no20s, "$20-bill" )
else:
    print( no20s, "$20-bills" )
```

Bits, binary switch

Relational Operators

Logical Operators

Teller Machine Revisited

Rock, Paper, Scissors

Exercises

Relational Operators

Operator	Meaning
==	equal to
!=	not equal to
<	less than
<=	less than or equal to
>	greater than
>=	greater than or equal to

Examples

```
if no20s == 1:
   print( no20s, "$20-bill" )
else:
   print( no20s, "$20-bills" )
```

```
amount = eval( input( "Amount to withdraw? " ) )
if amount > 400:
   print( "You are limited to $400 a week." )
   amount = 400
```

Examples

```
if no20s != 1:
   print( no20s, "$20-bills" )
else:
   print( no20s, "$20-bill" )
```

```
suffix = ""
if no20s != 1:
    suffix = "s"

print( "$20-bill"+suffix )
```

Coding Exercise

Recode the Teller-Machine program, so that

- **the output correctly displays"bill" or "bills"**
- **a number of bills of 0 is not displayed**
- **only an amount less than $400 is allowed**
- **only amounts multiples of $5 are allowed.**

(use tellerWithIfsPrep.py)

Exercises (Group 1)

http://cs.smith.edu/dftwiki/index.php/CSC111_Exercises_with_If_Statements_(Python_3)

```
def f0():
    a = 3
    b = 5
    c = 10

    if a < b:
        print( "statement 1" )
    else:
        print( "statement 2" )

    print( "f0 done!" )

f0()
```

Exercises (Group 1)

http://cs.smith.edu/dftwiki/index.php/CSC111_Exercises_with_If_Statements_(Python_3)

```
def f0():
    a = 30   # <== changed!
    b = 5
    c = 10

    if a < b:
        print( "statement 1" )
    else:
        print( "statement 2" )

    print( "f0 done!" )

f0()
```

Exercises (Group 1)

http://cs.smith.edu/dftwiki/index.php/CSC111_Exercises_with_If_Statements_(Python_3)

```
def f0():
    a = 30   # <== changed!
    b = 5
    c = 10
    if a < b:
        
    else:
        
  print( "f0 done!" )
```

Exercises (Group 1)

http://cs.smith.edu/dftwiki/index.php/CSC111_Exercises_with_If_Statements_(Python_3)

```
def f0():
    a = 3    # <== changed again!
    b = 5
    c = 10
    if a < b:
        if c == 10:
            print( "statement 1" )
        else:
            print( "statement 2" )
    else:
        print( "statement 3" )

    print( "f0 done!" )
```

Exercises (Group 1)

http://cs.smith.edu/dftwiki/index.php/CSC111_Exercises_with_If_Statements_(Python_3)

```
def f0():
    a = 3
    b = 1          # <== changed!
    c = 101        # <== changed!
    if a < b:
        if c == 10:
            print( "statement 1" )
        else:
            print( "statement 2" )
    else:
        print( "statement 3" )

    print( "f0 done!" )
```

Exercises (Group 1)

http://cs.smith.edu/dftwiki/index.php/CSC111_Exercises_with_If_Statements_(Python_3)

```
def f0():
    a = 3
    b = 1          # <== changed!
    c = 101        # <== changed!
    if a < b:
        if c == 10:
            print( "statement 1" )
        else:
            print( "statement 2" )
    else:
        print( "statement 3" )

    print( "f0 done!" )
```

Photo: https://www.orfit.com/blog/the-peripheral-nerves-a-quick-assessment-of-nerve-functioning-or-rock-paper-scissors/

Rock, Paper, Scissors

Rock, Paper, Scissors

R

R

P

P

S

S

Rock, Paper, Scissors

R

R

P

P

S

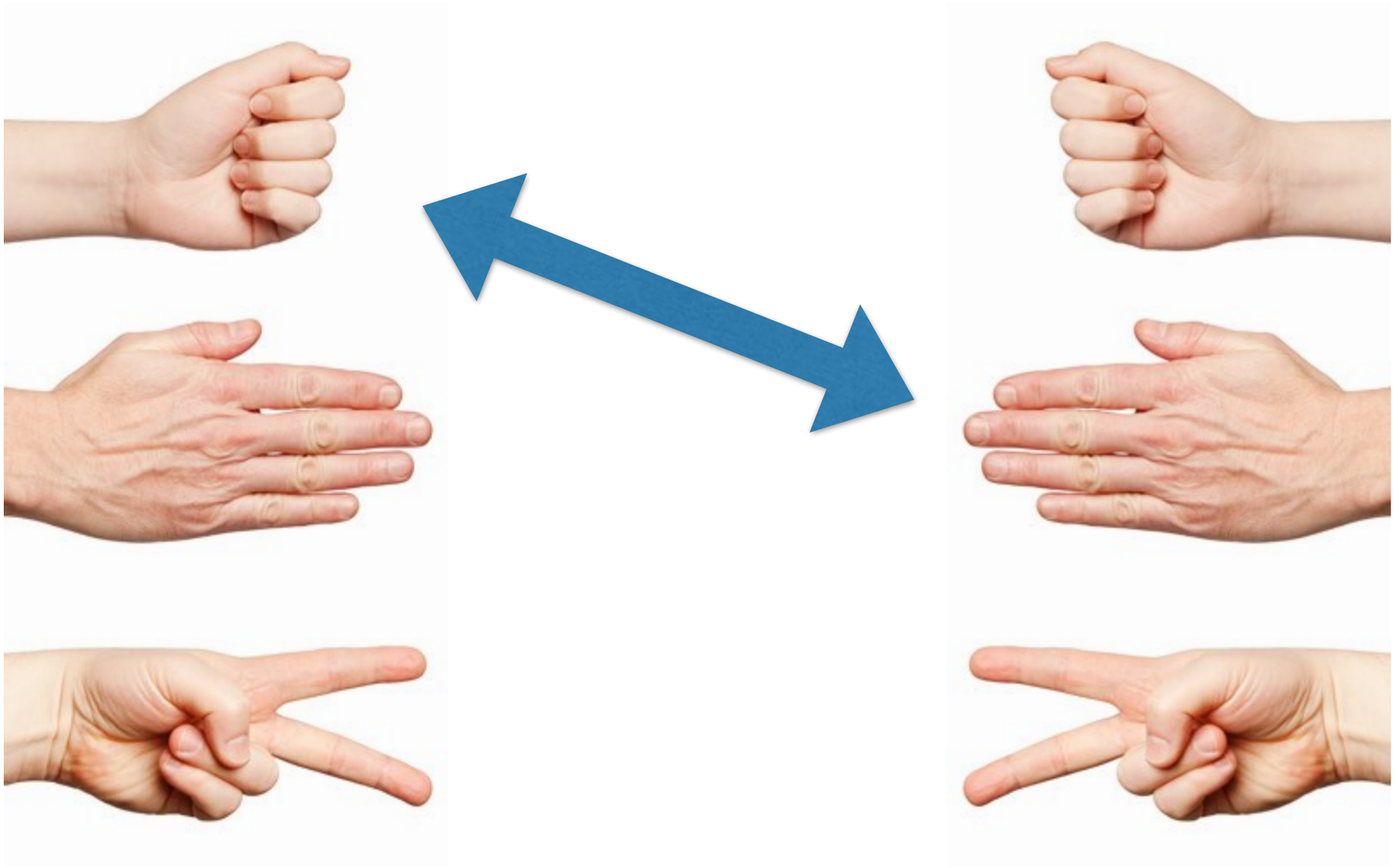
S

Rock, Paper, Scissors

R

P

S



R

P

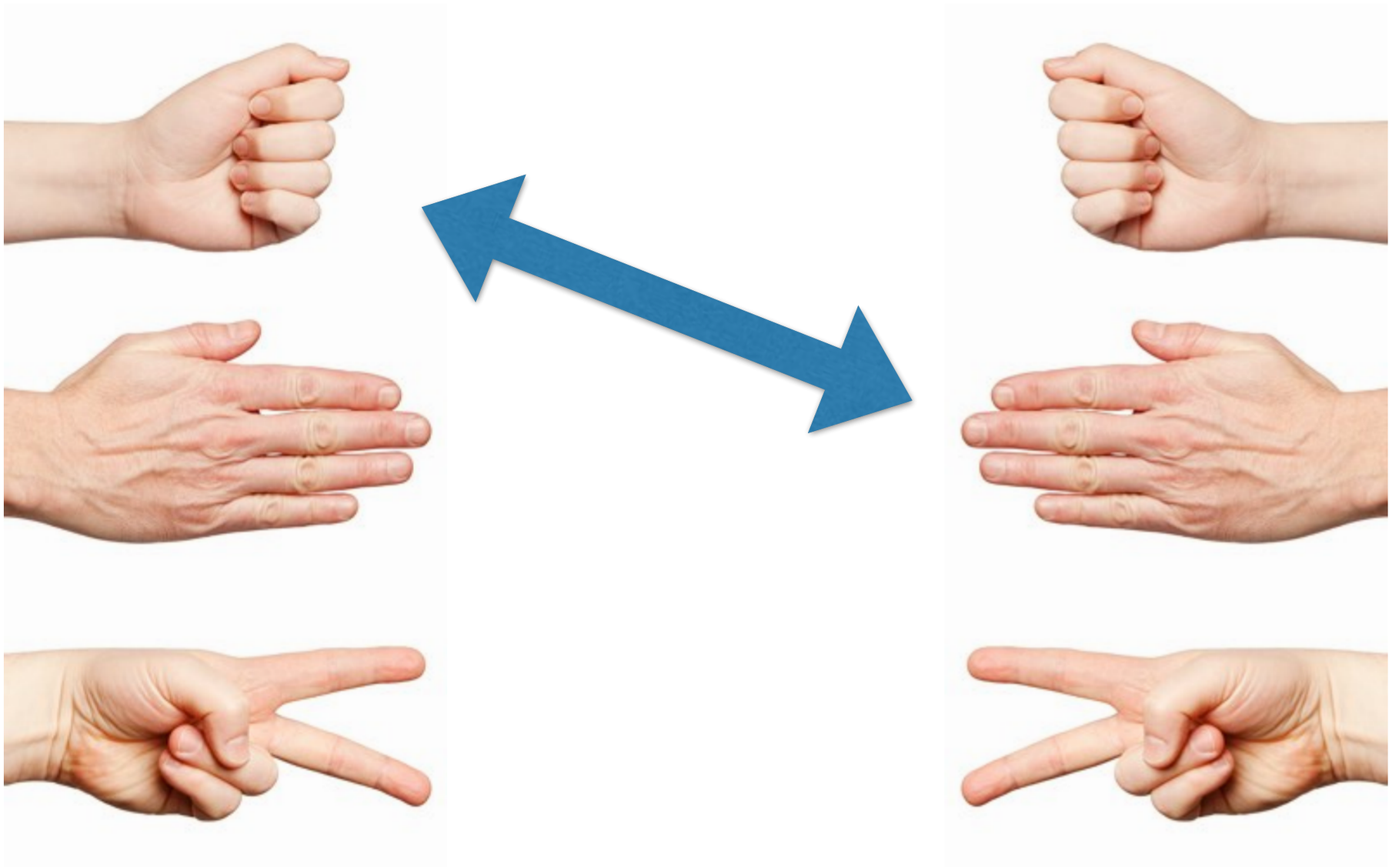
S

Rock, Paper, Scissors

R

P

S



R

P

S

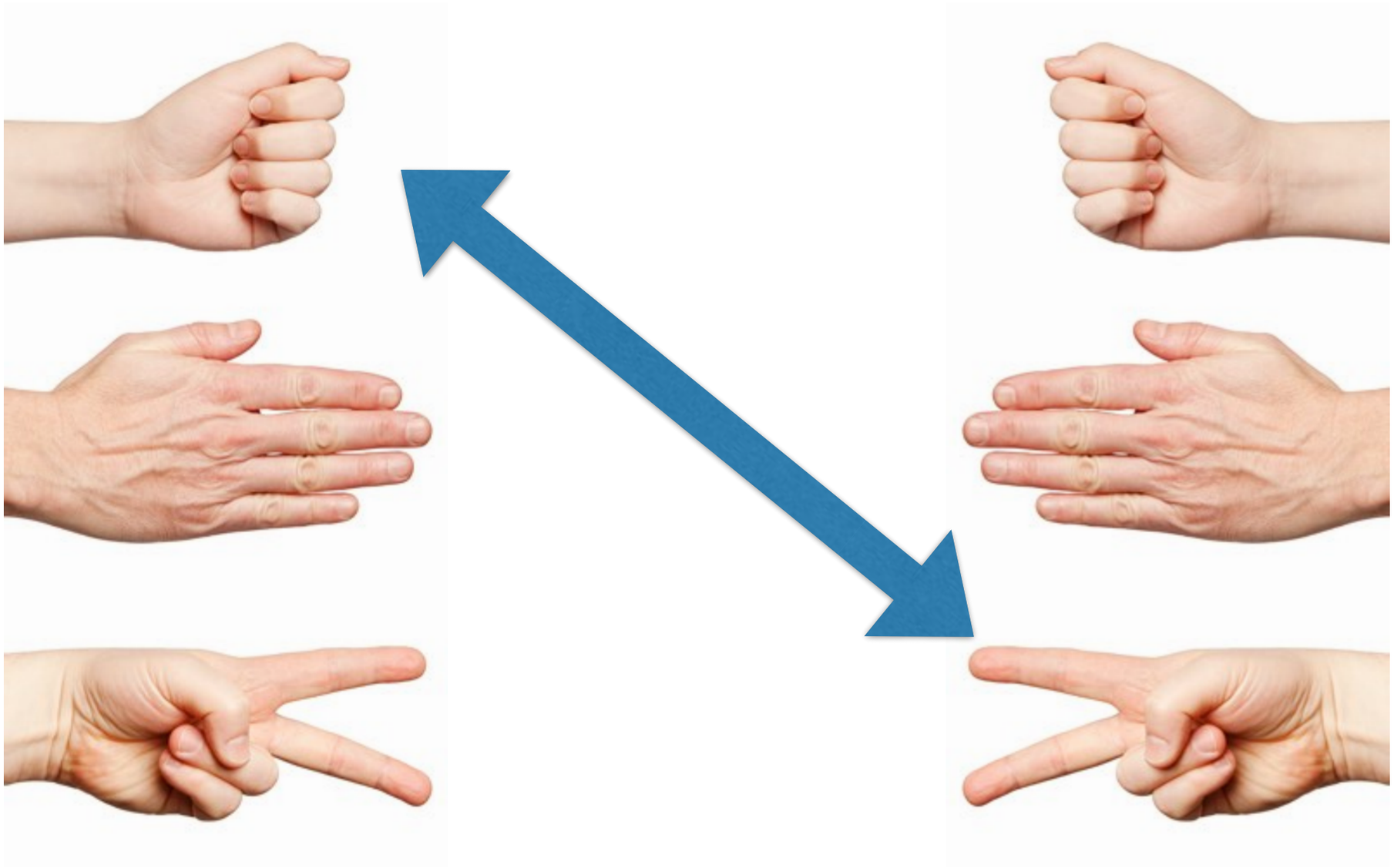
paper > rock

Rock, Paper, Scissors

R

P

S



R

P

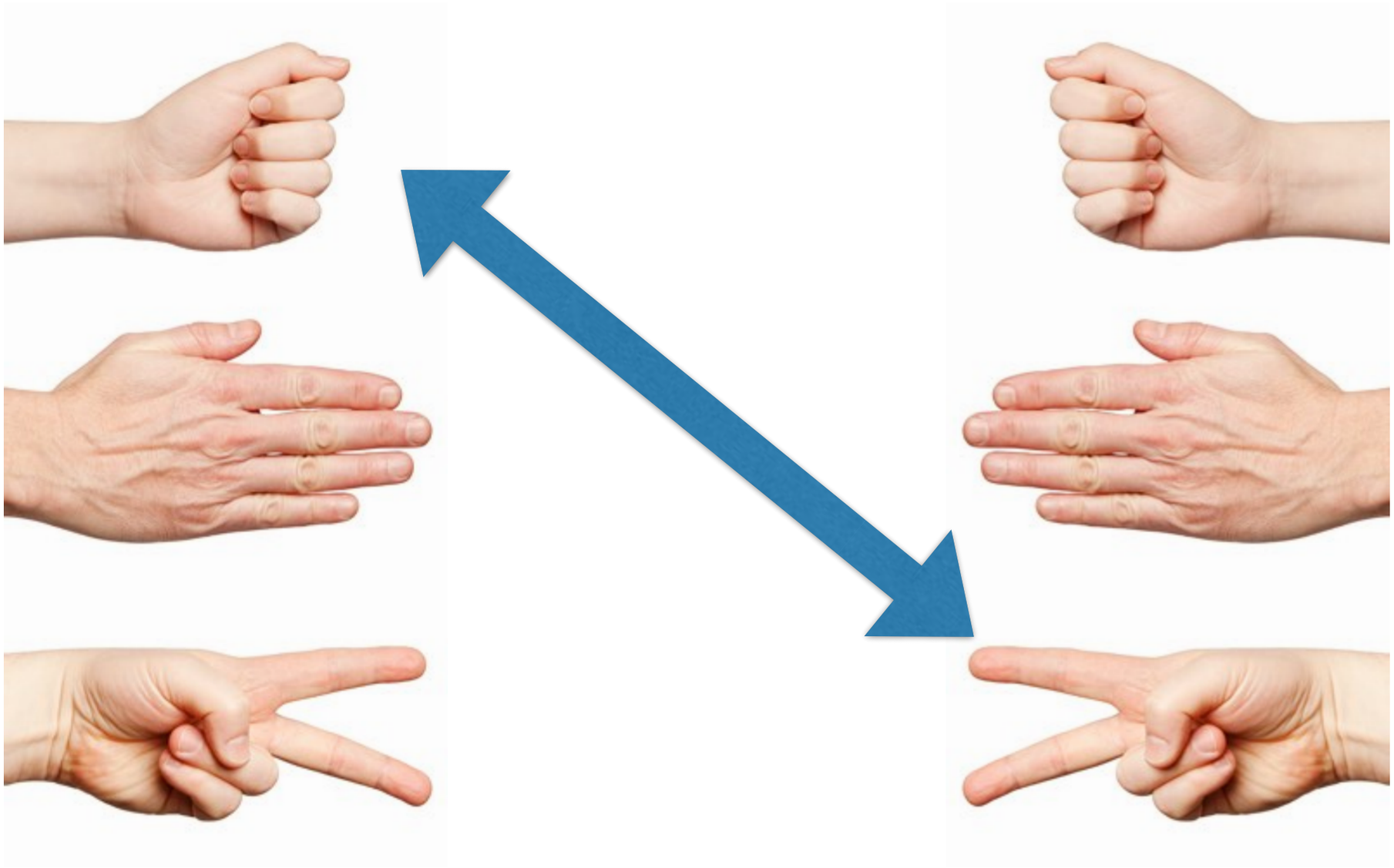
S

Rock, Paper, Scissors

R

P

S



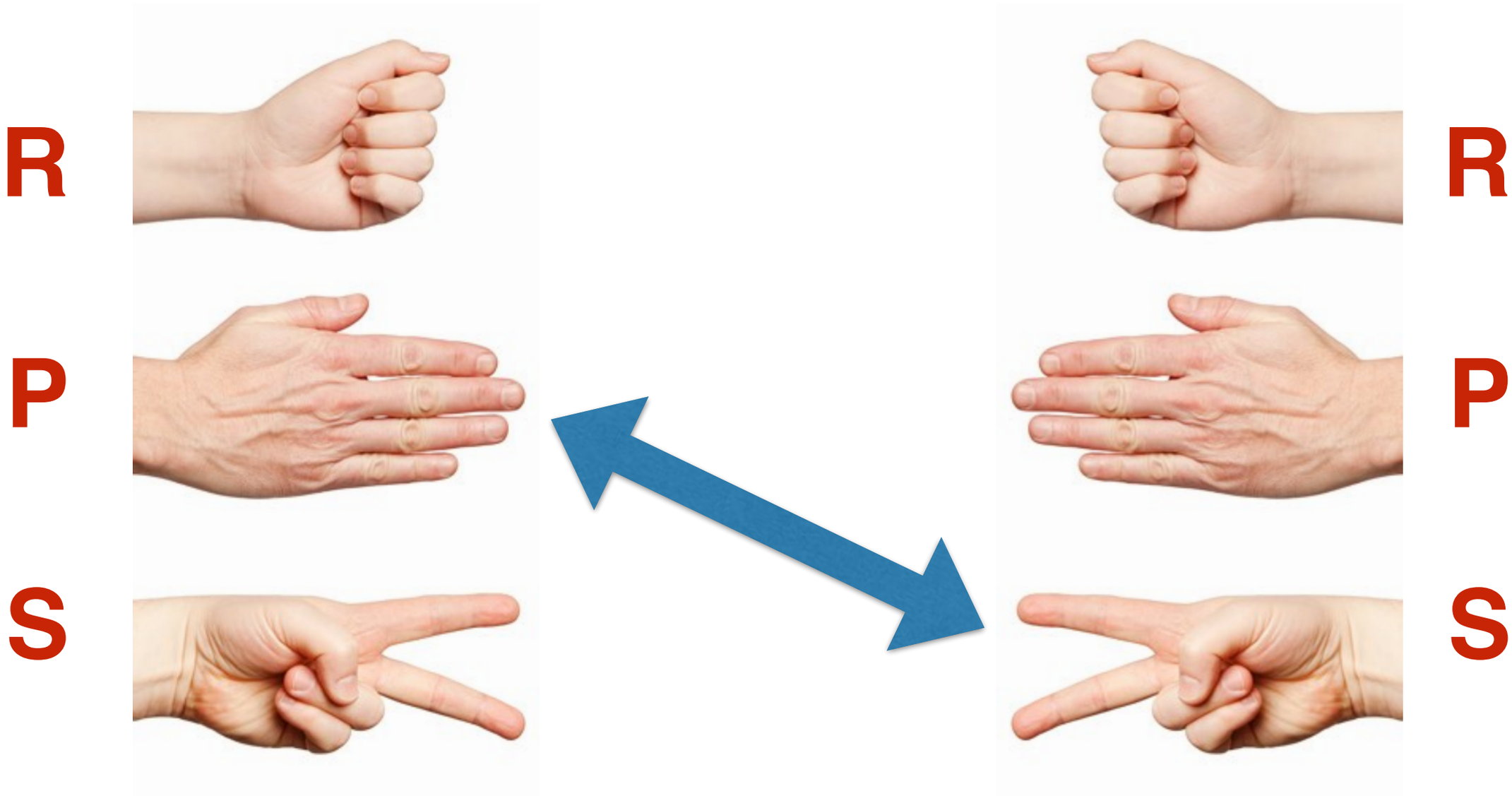
R

P

S

rock > scissors

Rock, Paper, Scissors

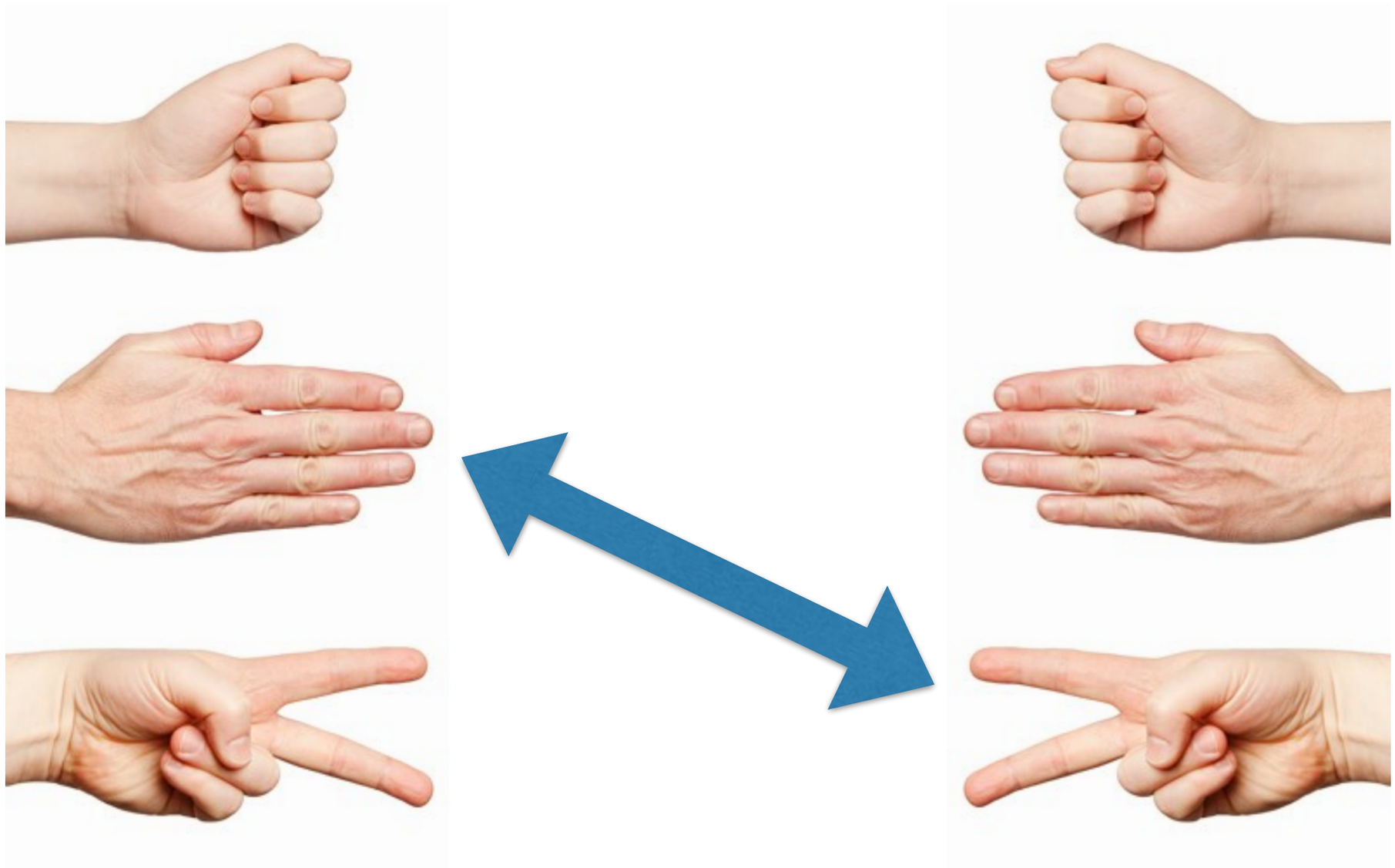


Rock, Paper, Scissors

R

P

S



R

P

S

scissors > paper

Exercise

- Write a game or Rock-Paper-Scissors
- Program prompts user for 2 letters
- First letter = Player 1
- Second letter = Player 2
- Program decides who wins
- Program keeps on going until some condition is true

Nifty Test

filterVowels.py - /Users/thiebaut/Desktop/Dropbox/111/filterVowels.py (3.5.4)

```
# filterVowels.py
# D. Thiebaut
# gets a string from user
# and keep only the vowels in the string

def main():
    answer = input( "> " ).strip().lower()
    valid = ""
    for letter in answer:
        if letter in [ 'a', 'e', 'i', 'o', 'u' ]:
            valid = valid + letter

    print( "valid letters = ", valid )

main()
```

Ln: 16 Col: 0

"Stopping" a For-Loop

stopForLoop.py - /Users/thiebaut/Desktop/Dropbox/111/stopForLoop.py (3.5.4)

```
# stopForLoop.py
# D. Thiebaut
# (there are better ways to do this,
# but when we don't know about the
# "break" statement, this will do!

def main():

    plays = [ "SS", "PP", "PS", "SP", "PR", "RR", "RR" ]

    count = 0
    for players in plays:
        if count < 3:
            print( "processing", players )

        count = count + 1

main()
```

Ln: 6 Col: 0