# More Zelle Graphics

User input

- Mouse click events with **getMouse()** <- Waits until something happens!
- Keyboard events with **getKey()** <- Waits until something happens!
- **checkMouse() and checkKey()** don’t block so you can use them in loops to check if an event occurs since the last time they were called

Colors are tuples of red,green,blue values

- Can refer to colors by tuple or by string
- See the assignment for details...

# More Zelle Graphics

Animation

Works like a flipbook where we repeatedly move objects, then draw

But! We must be careful of time for the animation to work correctly

**time.sleep(0.03)** # system call which pauses program for 0.03 seconds

**update(30)** # Zelle call which maintains 30 fps (smoother than sleep)!

# Exercise: Circle on mouse click

Draw a circle centered at the user's mouse click


helloMouse1.py — ~/CS21/cs21-devel/examples/inclass/w06 — Atom

File Edit View Selection Find Packages Help

helloGraphics.py | helloMouse1.py | helloMouse.py

```
from graphics import *

def main():
    # Create a window with width 600 and height 400
    # win is an instance of the class GraphWin
    # To create win, we call the class's constructor
    win = GraphWin("Hello Graphics", 600, 400)

    # Wait for a mouse click from the user
    # We call the method getMouse() contained in class GraphWin
    clickPoint = win.getMouse()

    # Print the location of the mouse click
    print(clickPoint)

    # Draw a circle centered on the user's mouse click
    c = Circle(clickPoint, 10)
    # Set a color for the circle and draw
    purple = color_rgb(127,33,199)
    c.setFill(purple)
    c.draw(win)

    # Wait for another mouse click
    win.getMouse()

    # close the window
    win.close()

main()
```

helloMouse1.py 18:42 LF I UTF-8 Python 0 files

# Exercise: Animate a circle

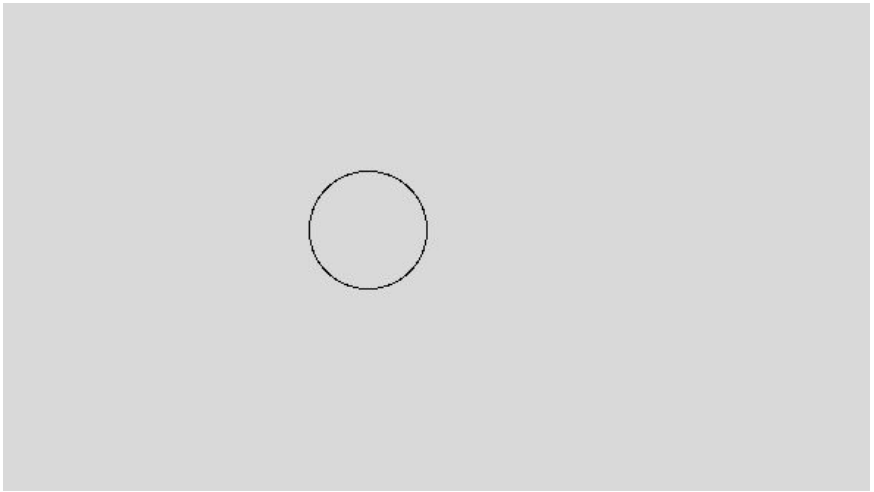
Animate a circle that moves from the left,middle of the screen to the right,middle of the screen.

What is the start point?

What is the end point?

What direction should the circle move in X and Y?

# Exercise: Animate a circle



helloMouse.py — ~/CS21/cs21-devel/examples/inclass/w06 — Atom

File Edit View Selection Find Packages Help

helloGraphics.py | helloMouse1.py | helloMouse.py

```
1  from graphics import *
2  import time
3
4  def main():
5      win = GraphWin("Hello Graphics", 600, 400)
6
7      # start on the left, middle of the screen
8      startPoint = Point(0,200)
9
10     # Create a circle with radius 40
11     animatedCircle = Circle(startPoint, 40)
12     animatedCircle.draw(win)
13
14     # Go across the screen in 100 steps, each step is 0.03 s
15     stepx = 600/100
16
17     # Don't go up or down -> Y should not change
18     stepy = 0
19
20     # store the circle's center in a variable
21     center = animatedCircle.getCenter()
22     # center has type Point. If we want its X value, we need
23     # to call the point's getX() method
24     while center.getX() < 600:
25         # Call the move() method to move the circle
26         # Moving the circle changes the location of its center
27         animatedCircle.move(stepx, stepy)
28         center = animatedCircle.getCenter()
29         # We can call either sleep or update
30         time.sleep(0.03)
31         #update(30)
32
33     win.getMouse()
34     win.close()
35
36 main()
```

helloMouse.py 28:5 LF N UTF-8 Python 0 file

# Exercise: bullseye

Ask the user for multiple mouse clicks and place a bullseye at each one using a function


bullseye.py — ~/CS21/cs21-devel/examples/inclass/w06 — Atom

File Edit View Selection Find Packages Help

helloGraphics.py | bullseye.py

```
from graphics import *
import time

def createCircle(win, center):
    """
    Create a Bullseye
    Create 3+ circles, each with the same center and
    decreasing radius
    BONUS: Make each circle a different color
    """
    circle1 = Circle(center, 30)
    circle1.setFill("green")
    circle1.draw(win)

    circle1 = Circle(center, 20)
    circle1.setFill("yellow")
    circle1.draw(win)

    circle1 = Circle(center, 10)
    circle1.setFill("orange")
    circle1.draw(win)

def main():
    win = GraphWin("Bullseye", 600, 400)

    for i in range(4):
        center = win.getMouse()
        createCircle(win, center)

    win.getMouse()
    win.close()

main()
```

bullseye.py 8:22 LF N UTF-8 Python 0 files