

# Myro Graphics

Douglas Blank

*Bryn Mawr College*

**Institute for Personal Robots in Education**

IPRE Winter Workshop

Atlanta, GA

January 7-9, 2011



**BRYN MAWR**

Microsoft  
**Research**



# Graphics

Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://bubo.brynmawr.edu/~dblank/cs110-pics/

Google

Google Calendar BMC Mail CS Mail Gmail - Inbox (1) IPRE People Genealogy Blogs Interests BMC ICOMP - Games

Facebook http...pl http...pl Gmail -... Tivo | ... Brain a... Paper... Google... Online ... ht...s/ AT&T T...

Aaron

The Face of All Grace

Alla

Anagha

Becky

Graphics Window

Emily

Madeline

Meena Seralathan

Samantha

Stephanie Viggiano's face

Jilly Forde

Teyvonia

Find: postscript

Next Previous Highlight all Match case

Done

# Games



YouTube game videos available at [cs.brynmawr.edu/games](https://cs.brynmawr.edu/games)

# Music

## Saxophone

A saxophone or similar wind instrument (*Saxofony* in Chuck)

- **setStiffness(*stiffness*)**: set reed stiffness ( $0.0 \leq \textit{stiffness} \leq 1.0$ )
- **setAperture(*aperture*)**: set reed aperture ( $0.0 \leq \textit{aperture} \leq 1.0$ )
- **setPressure(*pressure*)**: set pressure / volume ( $0.0 \leq \textit{pressure} \leq 1.0$ )
- **setVibrato(*vibratoFreq*, *vibratoGain*, *noiseGain*)**: set frequency and gain ( $\leq 1.0$ )
- **setBlowPosition(*position*)**: set blow position / lip stiffness ( $0.0 \leq \textit{positio} \leq 1.0$ )
- **startBlowing(*strength*)**: start blowing ( $0.0 \leq \textit{strength} \leq 1.0$ )
- **stopBlowing(*strength*)**: stop blowing ( $0.0 \leq \textit{strength} \leq 1.0$ )
- **setAttackRate(*seconds*)**: set rate of attack (sound's beginning) in seconds

## MoogSynthesizer

A Moog synthesizer (*Moog* in Chuck)

- **setFilterQ(*floatValue*)**: set filter's Q value ( $0.0 \leq \textit{floatValue} \leq 1.0$ )
- **setFilterSweepRate(*rate*)**: set filter sweep rate ( $0.0 \leq \textit{rate} \leq 1.0$ )
- **setVibrato(*freq*, *gain*)**: set frequency and gain of vibrato (*freq* in Hertz,  $0.0 \leq \textit{gain} \leq 1.0$ )
- **setAfterTouch(*afterTouch*)**: set aftertouch ( $0.0 \leq \textit{afterTouch} \leq 1.0$ )

## StruckBar

Struck bar instruments (*ModalBar* in Chuck)

## Orchestrating

After you get familiar with a single instrument, then you might want to play multiple instruments.

```
from myro import *
from myro.chuck import *

initChuck()

def playSaxophone():
    sax = Saxophone()
    sax.connect()
    sax.startBlowing(1)
    wait(1)
    sax.stopBlowing(1)

def playMandolin():
    mandolin = Mandolin()
    mandolin.connect()
    mandolin.pluck(1)
    wait(1)
```

You can test each one of those independently by simply running:

```
playSaxophone()
```

Once you have more than one instrument function written, you can play them together:

```
doTogether(playSaxophone, playMandolin)
```

# Instant Messaging and the Web

## Instant Messaging Interface

You can send and receive messages from other Myro users.

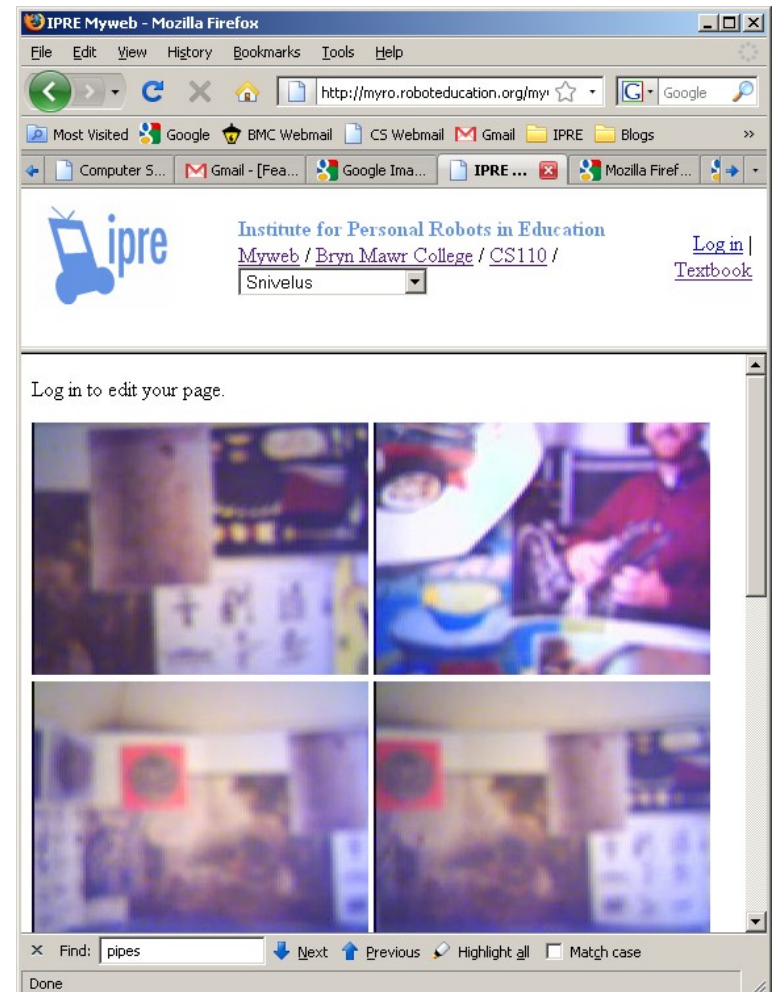
```
>>> chat = Chat("myname", "mypassword")
>>> chat.send("somebodyelse", "Hi, how are you?")
>>> chat.receive()
[("somebodyelse@myro.roboteducation.org", "I'm fine, thanks!")]
```

## Remote Robot Control

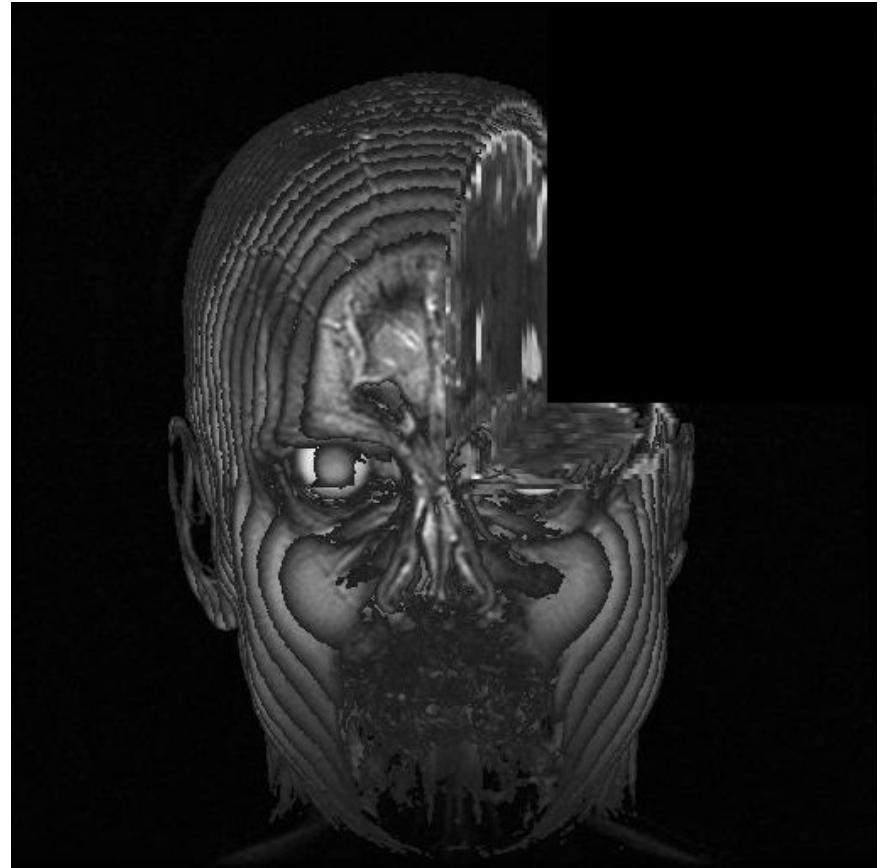
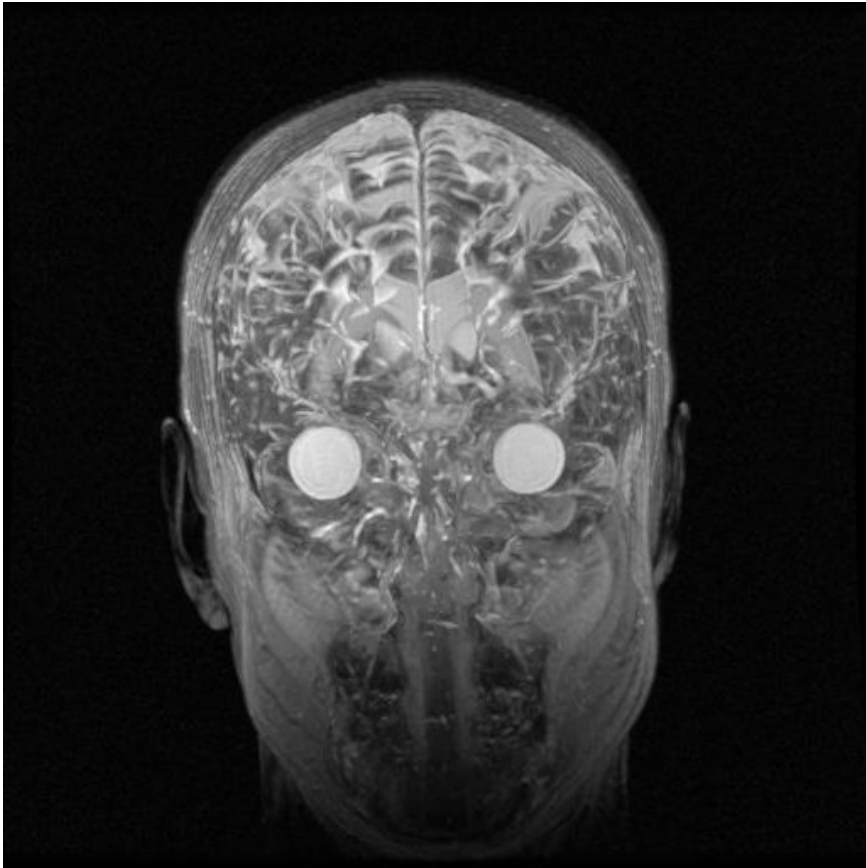
The robot that will be controlled:

```
>>> robot.initializeRemoteControl("mypassword")
>>> robot.processRemoteControl()
>>> []
>>> robot.processRemoteControlLoop() # threaded, infinite loop
>>>
```

The computer that will be the controller:



# Vision and Image Processing



# Graphics

- Different graphical objects:

- Window
- Points
- Lines
- Circles
- Polygons
- Text

```
from myro import *  
win = GraphWin()  
win.setBackground("red")
```

```
p = Point(100, 100)  
p.draw(win)
```

```
c = Circle(p, 50)  
c.setFill('green')  
c.draw(win)
```

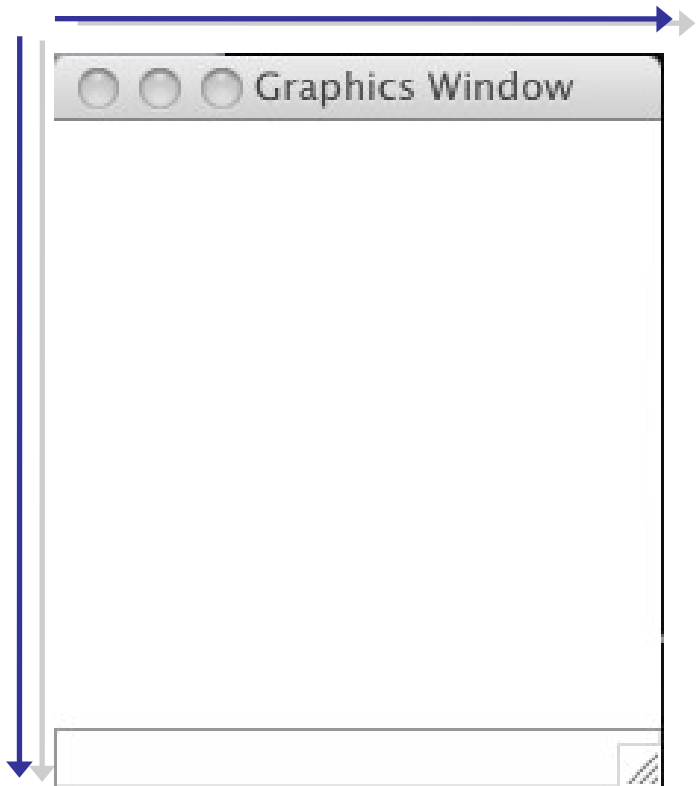
# GraphWin

- Main Graphics Window

```
win = GraphWin() #200x200 window
```

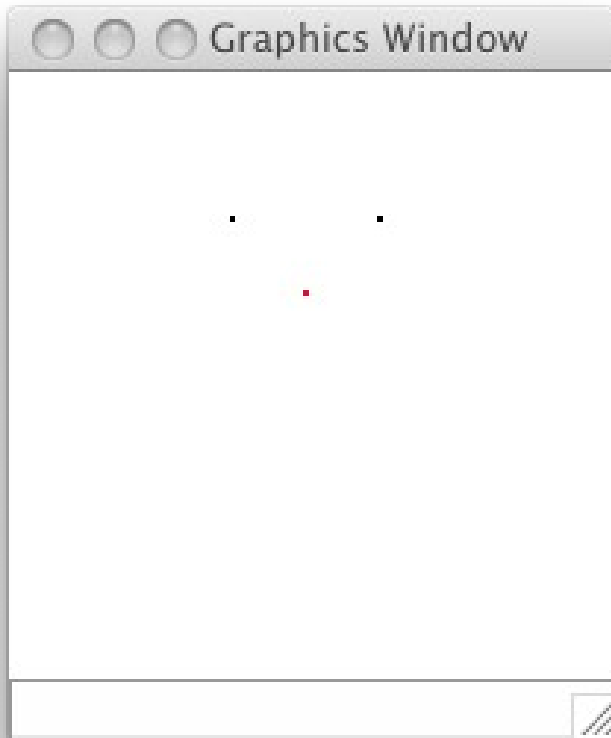
```
win.getWidth() # 200
```

```
win.getHeight() # 200
```



# Point

- An (X,Y) point



```
win = GraphWin() #200x200 window
```

```
p1 = Point(75, 50)
```

```
p1.draw(win)
```

```
p2 = Point(125, 50)
```

```
p2.draw(win)
```

```
p3 = Point(100, 75)
```

```
p3.setFill("red")
```

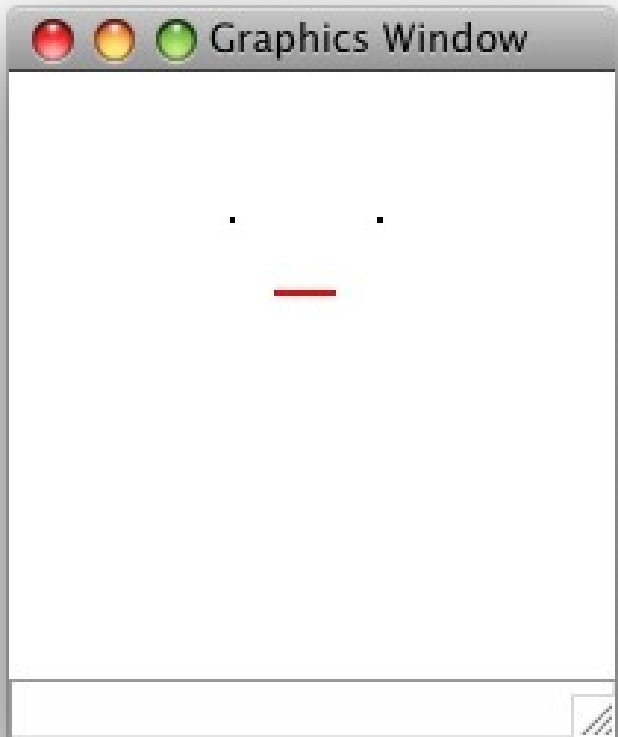
```
p3.draw(win)
```

```
p1.getX()
```

```
p1.getY()
```

# Points to Lines?

- A series of points



```
win = GraphWin() #200x200 window
```

```
for i in range(90, 110):
```

```
    p = Point(i, 75)
```

```
    p.setFill('red')
```

```
    p.draw(win)
```

```
p3 = Point(90, 75)
```

```
p4 = Point(110, 75)
```

```
l = Line(p3, p4)
```

```
l.setFill('red')
```

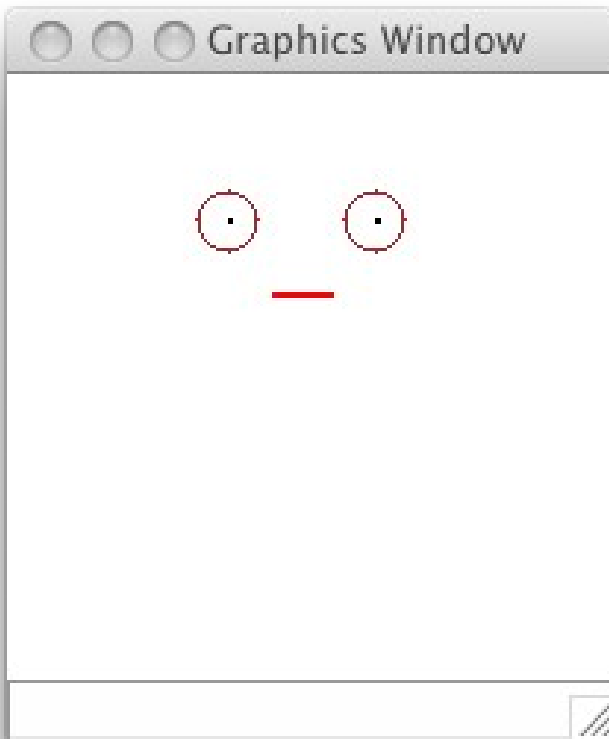
```
l.draw(win)
```

# Circles

- An (X,Y) point and a radius

```
c1 = Circle(p1, 10)
c1.setOutline('brown')
c1.draw(win)
```

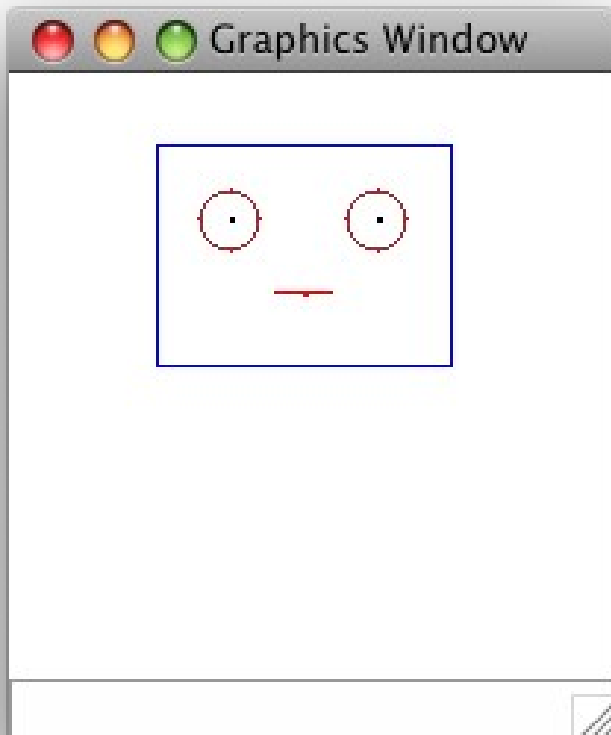
```
c2 = Circle(p2, 10)
c2.setOutline('brown')
c2.draw(win)
```



# Rectangles

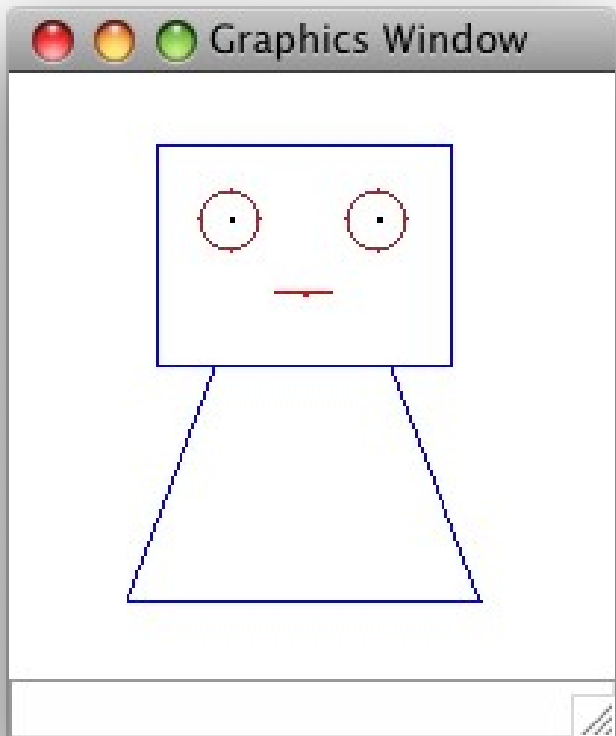
- The two corners as (X,Y) points

```
p5 = Point(50, 25)
p6 = Point(150, 100)
r = Rectangle(p5, p6)
r.setOutline('blue')
r.draw(win)
```



# Polygons

- A series of (X,Y) points



```
p6 = Point(70, 100)
p7 = Point(130, 100)
p8 = Point(160, 180)
p9 = Point(40, 180)
```

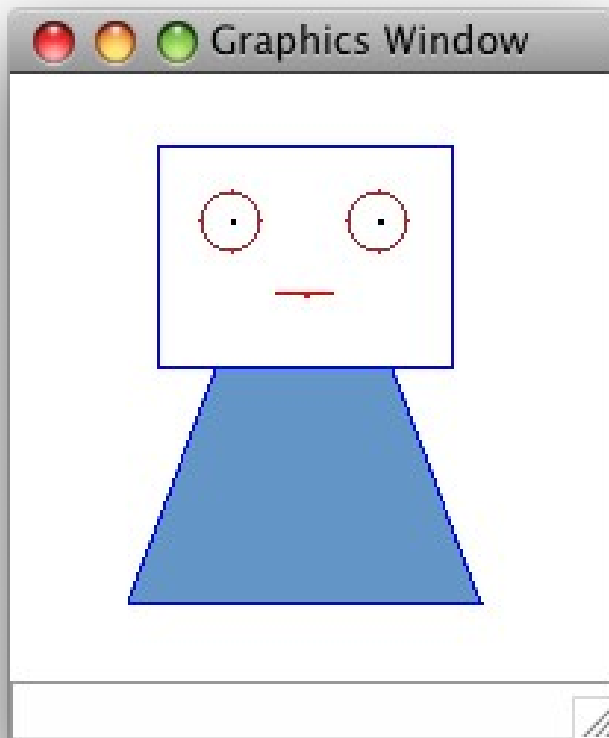
```
body = Polygon(p6, p7, p8, p9)
body.setOutline('blue')
body.draw(win)
```

# RGB Colors

- RGB values (0-255)

```
p6 = Point(70, 100)
p7 = Point(130, 100)
p8 = Point(160, 180)
p9 = Point(40, 180)
```

```
body = Polygon(p6, p7, p8, p9)
body.setOutline("blue")
bodyColor = color_rgb(100, 150, 200)
body.setFill(bodyColor)
body.draw(win)
```



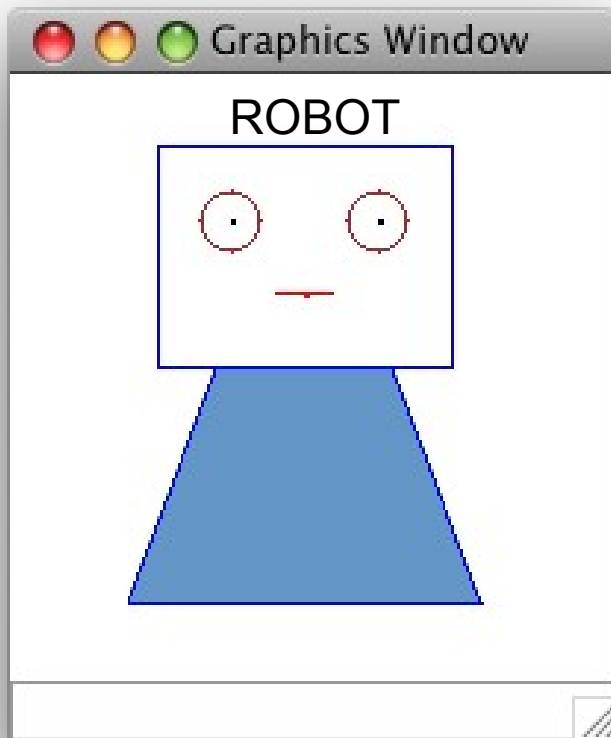
# Text

- A point & string

```
p10 = Point(70, 20)
```

```
t = Text(p10, "ROBOT")
```

```
t.draw(win)
```



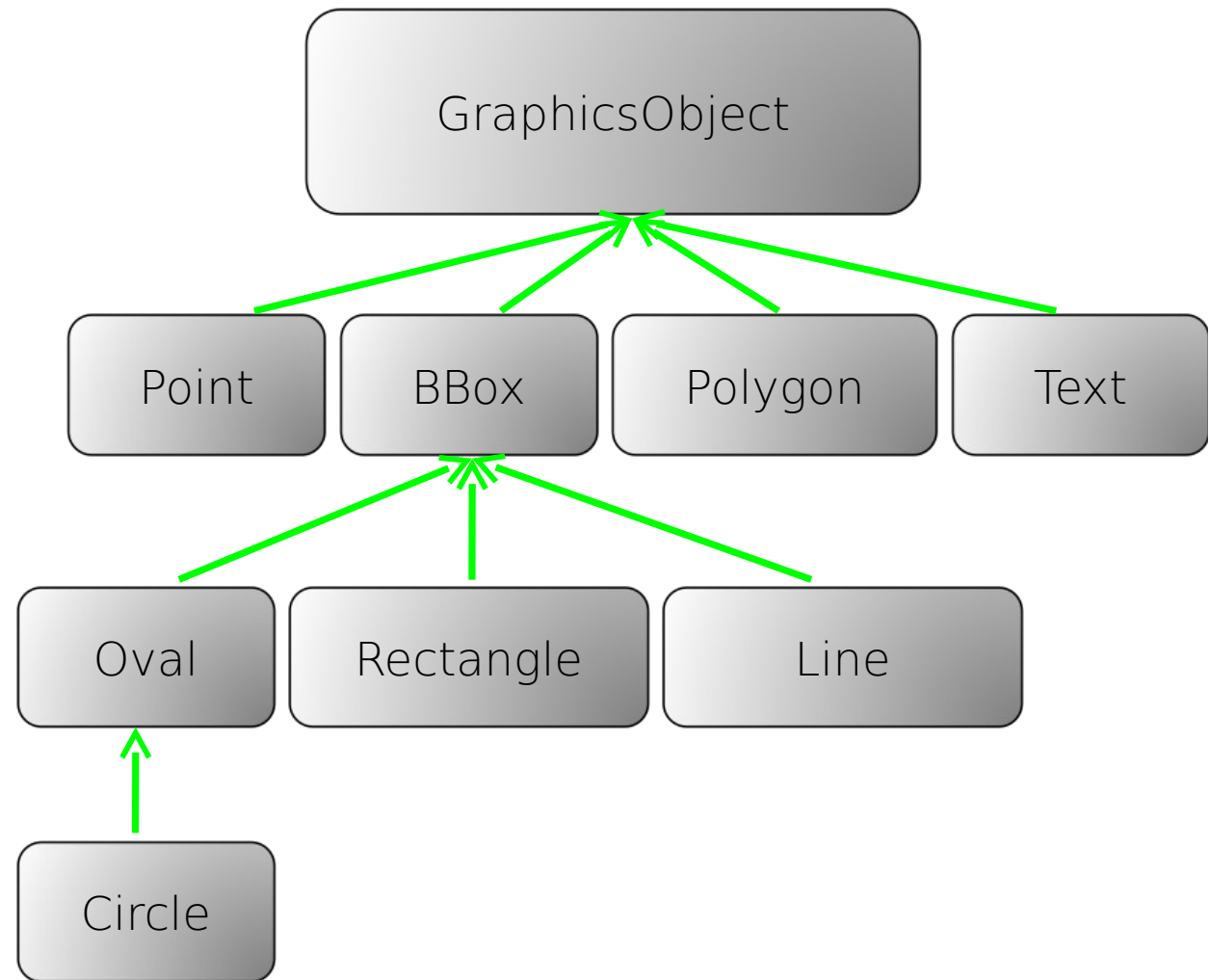
# Shapes - Inheritance

- All shapes inherit from a common shape

- Inherit common methods

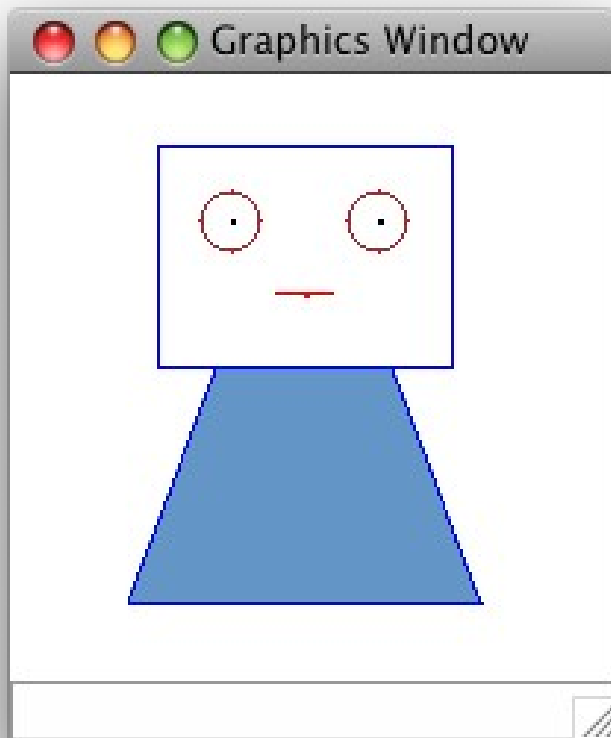
- `setFill(color)`
- `setOutline(color)`
- `getCenter()`
- `move(dx, dy)`

- `help(Circle)`



# move(dx,dy)

- Move any shape or point by dx, dy



```
while True:
```

```
    x = random.uniform(-1, 1)
```

```
    y = random.uniform(-1, 1)
```

```
    p1.move(x, y)
```

```
    p2.move(x, y)
```

```
    y = random.uniform(-2, 2)
```

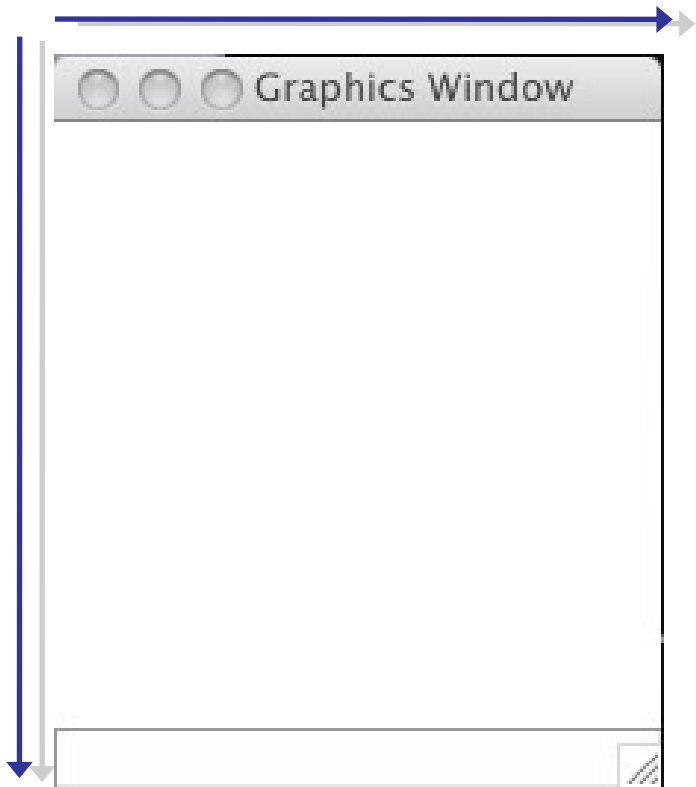
```
    r.move(0, y)
```

```
    body.move(0, y)
```

```
    wait(.1)
```

# Revisiting GraphWin

- Main Graphics Window



```
win = GraphWin() #200x200 window
```

```
win = GraphWin("drawing", 400, 400)
```

```
win = GraphWin(title, width, height)
```

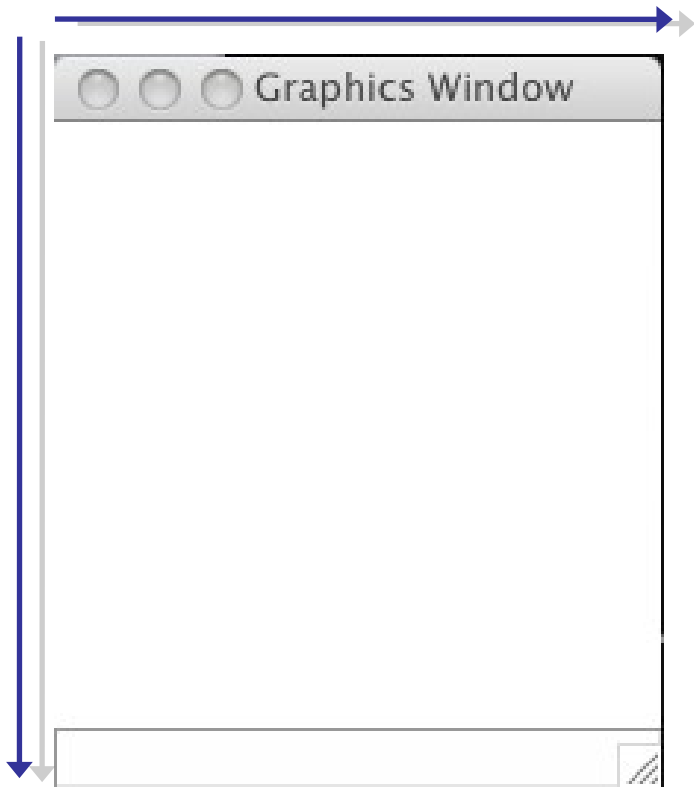
```
#bottom left, and upper right coords  
win.setCoords(-1, -1, 1, 1)
```

```
win.plot(x, y)
```

# Getting Mouse Input

- Main Graphics Window

```
win = GraphWin() #200x200 window  
p = win.getMouse()  
p.draw(win)
```



# Multiple Robots - Objects

- You can control multiple robots from one python shell
- Use the “object” interface to the robot
- **Methods**
  - functions on objects

```
alice = Scribbler('com30')  
bob = Scribbler('com31')
```

```
alice.forward(0.75, 1)  
bob.beep(1, 440)
```

```
apic = alice.takePicture()  
bpic = bob.takePicture()
```

```
computer.beep(1, 440)  
computer.playSong()
```