“This is a medical robot, guided by a skilled surgeon and designed to get to places doctors are unable to reach without opening a patient up.

It is still only a prototype and has not yet been used on real patients - only in the lab.

But its designers, from OC Robotics in Bristol, are convinced that once ready and approved, it could help find and remove tumours.”

http://www.bbc.co.uk/news/health-19653105
• Lab?

Writing all at once
vs.

Testing as you go
>>> ### Open a graphics window
>>> win = GraphWin('Shapes')

>>> ### Draw a red circle centered at point (100, 100) with radius 30
>>> center = Point(100, 100)
>>> circ = Circle(center, 30)
>>> circ.setFill('red')
>>> circ.draw(win)

>>> ### Put a textual label in the center of the circle
>>> label = Text(center, "Red Circle")
>>> label.draw(win)

>>> ### Draw a square using a Rectangle object
>>> rect = Rectangle(Point(30, 30), Point(70, 70))
>>> rect.draw(win)

>>> ### Draw a line segment using a Line object
>>> line = Line(Point(20, 30), Point(180, 165))
>>> line.draw(win)

>>> ### Draw an oval using the Oval object
>>> oval = Oval(Point(20, 150), Point(180, 199))
>>> oval.draw(win)
Using Graphical Objects

• Each object is an instance of some class, and the class describes the properties of the instance

• If we say that Augie is a dog, we are actually saying that Augie is a specific individual in the larger class of all dogs. Augie is an instance of the dog class

• Another way to think about it:
  – Class = blueprint for a house
  – Instance = house built from that blueprint
Using Graphical Objects

• Draw method: uses information about center and radius from instance variables
Getting Mouse Clicks

# triangle.pyw
# Interactive graphics program to draw a triangle

from graphics import *

def main():
    win = GraphWin("Draw a Triangle")
    win.setCoords(0.0, 0.0, 10.0, 10.0)
    message = Text(Point(5, 0.5), "Click on three points")
    message.draw(win)

    # Get and draw three vertices of triangle
    p1 = win.getMouse()
    p1.draw(win)
    p2 = win.getMouse()
    p2.draw(win)
    p3 = win.getMouse()
    p3.draw(win)
Getting Mouse Clicks

# Use Polygon object to draw the triangle
triangle = Polygon(p1,p2,p3)
triangle.setFill("peachpuff")
triangle.setOutline("cyan")
triangle.draw(win)

# Wait for another click to exit
message.setText("Click anywhere to quit.")
win.getMouse()

main()
Handling Textual Input

• Input can be through mouse clicks
• Also an Entry object for keyboard input
• The Entry object draws a box on the screen that can contain text. It understands `setText` and `getText`, but now input can be edited
Handling Textual Input
Handling Textual Input

```python
# convert_gui.pyw
# Program to convert Celsius to Fahrenheit using a simple
#   graphical interface.

from graphics import *

def main():
    win = GraphWin("Celsius Converter", 300, 200)
    win.setCoords(0.0, 0.0, 3.0, 4.0)

    # Draw the interface
    Text(Point(1,3), "   Celsius Temperature:").draw(win)
    Text(Point(1,1), "Fahrenheit Temperature:").draw(win)
    input = Entry(Point(2,3), 5)
    input.setText("0.0")
    input.draw(win)
    output = Text(Point(2,1),"")
    output.draw(win)
    button = Text(Point(1.5,2.0),"Convert It")
    button.draw(win)
    Rectangle(Point(1,1.5), Point(2,2.5)).draw(win)
```

Python Programming, 2/e
Handling Textual Input

# wait for a mouse click
win.getMouse()

# convert input
celsius = eval(input.getText())
fahrenheit = 9.0/5.0 * celsius + 32

# display output and change button
output.setText(fahrenheit)
button.setText("Quit")

# wait for click and then quit
win.getMouse()
win.close()

main()
Handling Textual Input

![Celsius Converter](image)

Celsius Temperature: 28.0

Fahrenheit Temperature: 82.4
Handling Textual Input

• When run, this program produces a window with an entry box for typing in the Celsius temperature and a button to “do” the conversion
  – The button is for show only
  – We’re just waiting for a mouse click anywhere in the window (other than the text box)
Handling Textual Input

• Initially, the input entry box is set to contain “0.0”

• User can delete this value and type in another value

• The program pauses until the user clicks the mouse – we don’t care where so we don’t store the point
Handling Textual Input

• The input is processed in three steps:
  – The value entered is converted into a number with `eval`
  – This number is converted to degrees Fahrenheit
  – This number is then converted to a string and formatted for display in the output text area
• Have button actually do something?
Simple paint program

• **DrawButtons( win )**
  – Called during program setup
  – Draws buttons to screen
  – No return value

• **ButtonClicked( x1, y1, x2, y2, p )**
  – Input = location of click
  – returns true iff button defined by (x1,y1), (x2,y2) has been clicked

• **ChangeColor( p, color )**
  – Called when user clicks at point p
  – Input = location of click, current color
  – Returns new color value if button was clicked, or old value if no color was clicked

• **HandleClick( p, pOld, color )**
  – Called when user clicks at point p (pOld was last place clicked)
  – Handle a button press, if it was clicked
  – Otherwise, draw a line from pOld to p using current color