Informal quiz (discuss with a partner)

1) \textit{c} is an _______ of the \text{Circle} _______.

2) \texttt{GraphWin(..), Point(..), and Circle(..) are all _______.}

3) \texttt{width/2, height/2, “white”, “blue” are all _______.}

4) \texttt{setFill(..), setOutline(..), and draw(..) are all _____ not _____}.
Informal quiz (discuss with a partner)

1) \( c \) is an \textit{instance} of the \texttt{Circle} class.

2) \texttt{GraphWin(..), Point(..), and Circle(..)} are all ________.

3) \texttt{width/2, height/2, "white", "blue"} are all ________.

4) \texttt{setFill(..), setOutline(..), and draw(..)} are all _____ not not ________.

```python
width = 600
height = 600
win = GraphWin("Random Circles", width, height)
win.setBackground("white")

p = Point(width/2, height/2)
c = Circle(p, 10)
c.setFill("blue")
c.setStroke("blue")
c.draw(win)
```
Informal quiz (discuss with a partner)

1) $c$ is an instance of the Circle class.

2) GraphWin(..), Point(..), and Circle(..) are all constructors.

3) $\text{width}/2$, $\text{height}/2$, "white", "blue" are all ________.

4) setFill(..), setOutline(..), and draw(..) are all _____ not not _____.

```python
width = 600
height = 600
win = GraphWin("Random Circles", width, height)
win.setBackground("white")

p = Point(width/2, height/2)
c = Circle(p, 10)
c.setFill("blue")
c.setFill("blue")
c.draw(win)
```
Informal quiz (discuss with a partner)

1) c is an instance of the Circle class.

2) GraphWin(..), Point(..), and Circle(..) are all constructors.

3) width/2, height/2, “white”, “blue” are all arguments/parameters.

4) setFill(..), setOutline(..), and draw(..) are all ______ not ______.
Informal quiz (discuss with a partner)

1) \texttt{c} is an \textit{instance} of the \texttt{Circle} \texttt{class}.

2) \texttt{GraphWin(..)}, \texttt{Point(..)}, and \texttt{Circle(..)} are all \textit{constructors}.

3) width/2, height/2, “white”, “blue” are all \textit{arguments/parameters}.

4) \texttt{setFill(..)}, \texttt{setOutline(..)}, and \texttt{draw(..)} are all \textit{methods} not \textit{functions}.

```python
width = 600
g\texttt{height} = 600
\texttt{win} = \texttt{GraphWin("Random Circles", width, height)}
\texttt{win.setBackground("white")}

\texttt{p} = \texttt{Point(width/2, height/2)}
\texttt{c} = \texttt{Circle(p, 10)}
\texttt{c.setFill("blue")}
\texttt{c.setOutline("blue")}
\texttt{c.draw(win)}
```
Outline Oct 4:

• Recap OOP vocabulary
• Continue graphics
• User clicks
• Getters and setters
• Moving box program (box.py)
• Falling snow program (snow.py)

Notes

• Quiz 2 this Friday (let me know if you have conflicts)
• Lab 4 due Saturday night
• Office Hours 3-5pm on Friday (or by appointment)
Continue Graphics
GraphWin class

- **GraphWin**(title, width, height) – constructs a new graphics window (default width and height are both 200)
- **setBackground**(color) – set the background color
- **close()** – closes the window
- **getMouse()** – waits for the user to click, returns the click position as a Point
- **checkMouse()** – does not wait for the user to click, returns the click position as a Point, or None if no position clicked
Methods for all Graphics Objects

- **setFill(color)** – sets the interior color of an object
- **setOutline(color)** – sets the outline color of an object
- **setWidth(pixels)** – sets the outline width (doesn’t work for Point)
- **draw(window)** – draws the object on the given window
- **undraw()** – removes the object from a graphics window
- **move(dx,dy)** – moves the object dx in the x direction and dy in the y direction
- **clone()** – returns a duplicate (new copy) of the object
Rectangle class

- `Rectangle(point1, point2)` – constructs a rectangle with opposite corners at the given points (upper left, lower right)
- `getCenter()` – returns the center point
- `getP1()`, `getP2()` – returns a clone of the corner point
User clicks and getters

- `win.getMouse()` waits for the user to click
- It returns the user’s click as a **Point**
- We can use that **Point** later on or extract the x and y coordinates using a *getter*

```python
click = win.getMouse()
print(click)
x = click.getX()  # getter for x coordinate
y = click.getY()  # getter for y coordinate
print(x_click, y_click)

c = Circle(click, 10)
center = c.getCenter()  # getter (what is the type of center?)
```
Programs for today
Work with a partner on one computer!

- *Pair programming* is frequently used in upper level CS classes and afterward in industry/academia
- One person is the *driver* at the keyboard (typing)
- The other person is the *navigator* who is providing advise, feedback, etc.
- Switch frequently between roles, email code at the end of class

- cs21/inclass/week05/box.py (first)
- cs21/inclass/week05/snow.py (second)