Objectives

- Exam Questions
- Problem solving using Graphics API

Exam Notes

- Given API for problems involving OO programming
- If you missed class, I have extra handouts
  - Practice for exam
- Lab advice
  - Do last two lab problems with OO programming
  - Then, finish up Craps
  - Extra credit problem: another practice on while loops
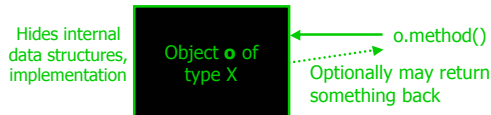   - Output: only up to 100 points

Object-Oriented Programming

- Objects **combine** data and methods together

Provides **interface** (the methods) that users interact with

Hides internal data structures, implementation

Object **o** of type X

o.method()

Optionally may return something back

Use an Application Programming Interface (**API**) to interact with a set of classes.

Benefits of Object-Oriented Programming

- Abstraction
  - Hides details of underlying implementation
  - Easier to change implementation
- Easy reuse of code
- Collects related data/methods together
  - Easier to reason about data

OO Terminology Summary

Term	Definition	Examples
Class	A data type. Defines the data and operations for members of the class	string, TV, GraphWin
Object	An instance of a specific class	animal, myTV, window
Method	Operations you can call on an object	setBackground(<color>), getWidth()
Constructor	Special method to create an object of a certain type/class	GraphWin(), str(1234)

What Does This Code Do?

- Use OO terminology previously defined

```
from graphics import *
```
Constructor

GraphWin object

Also known as an **instance of** the **GraphWin** class

```
win = GraphWin("My Circle", 100, 100)
c = Circle(Point(50,50), 10)
c.draw(win)
win.getMouse()
```

Method called on GraphWin object

1

A GraphWin Object's Canvas

(0,0)

X horizontal axis

origin

Coordinates are specified as (x,y)

Y
v
e
r
t
i
c
a
l

a
x
i
s

What are the coordinates for these points?

(200,200)
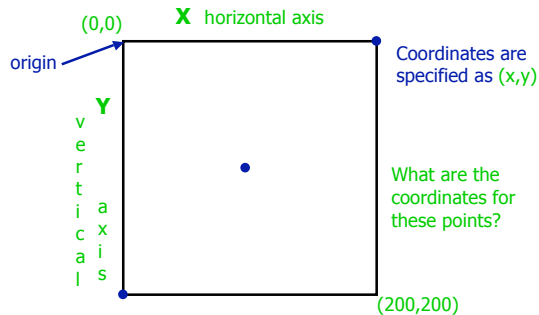
Getting Input from the User

- <GraphWin>.getMouse()
  - ➢ Returns the user's mouse click as a **Point** object
- Entry objects
  - ➢ Get text from user

Problem

- Create a program where the user tells you where to draw a line
  - ➢ What do you need from the user?
  - ➢ What do you need to create a line?

Problem: Circle Shift

- Move a circle to the position clicked by the user
  - ➢ Repeat five times

Animation

- Use combinations of the method **move** and the function **sleep**
  - ➢ Need to **sleep** so that humans can see the graphics moving
  - ➢ Computer would process the **move**s too fast!
- **sleep** is part of the **time** module
  - ➢ takes a float representing *seconds* and pauses for that amount of time

Example of Animation

- From last semester's class

rocket.py

Problem: Animate Moving to User Click

- In X steps, move from the circle's current location to the location clicked by user

animate2.py
fenway.py