# CS 329E
# Elements of Mobile Computing

Fall 2017
University of Texas at Austin

Lecture 6

# Agenda

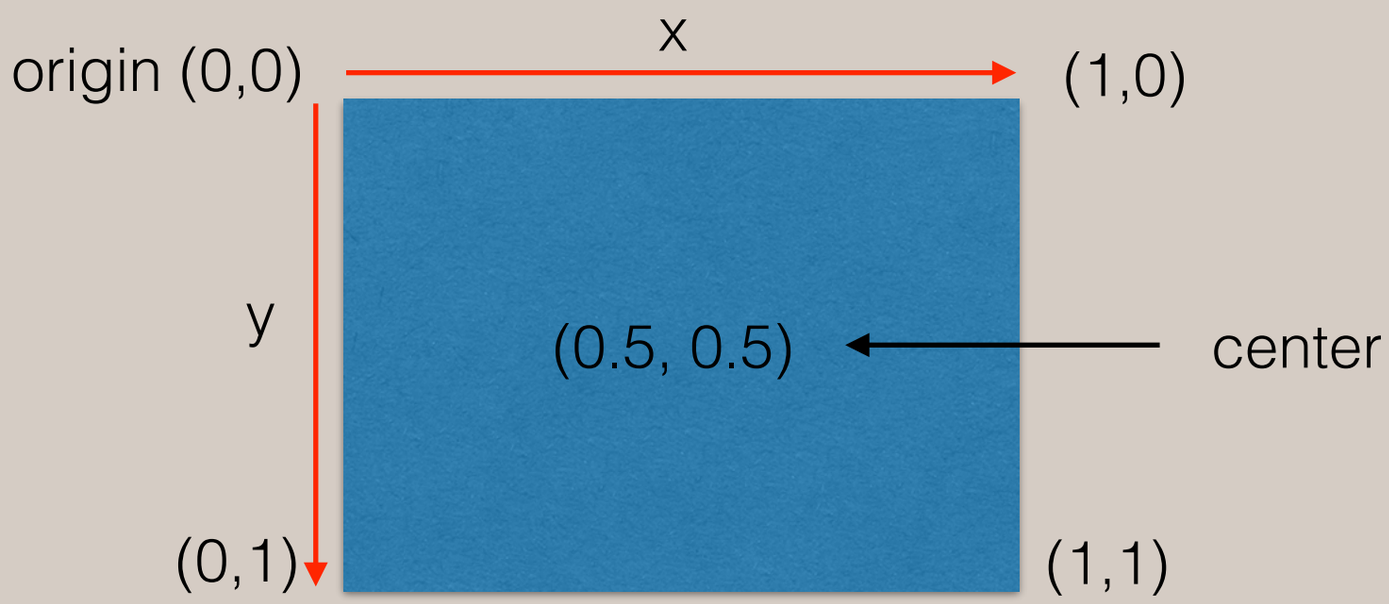
- Coordinate Systems
- View Frame and Bounds properties
- Other View properties
- Project
- Homework 3

# Coordinate Systems

# Coordinate Systems

- A coordinate system is a system which uses one or more numbers, or coordinates, to uniquely determine the position of a point in space.
- The components of a coordinate are real numbers.
- Coordinates can be positive or negative.

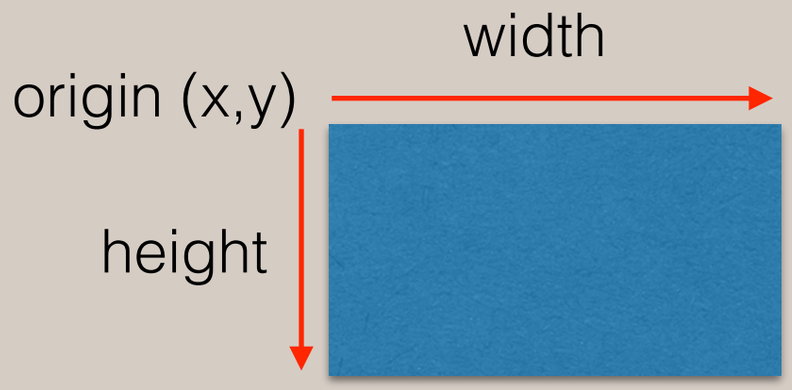
## iOS (normalized) 2D Coordinate System



# View Frame and Bounds

# View Frame and Bounds

- Every UI element derives from UIView.
- Frame and Bounds are properties of a UIView and, therefore, every UI element.
- Frame and Bounds are structures.
- The structure is a CGRect and consists of 4 floats:
  - origin (top-left point):
    - x, y
  - size:
    - width, height



# View Frame and Bounds

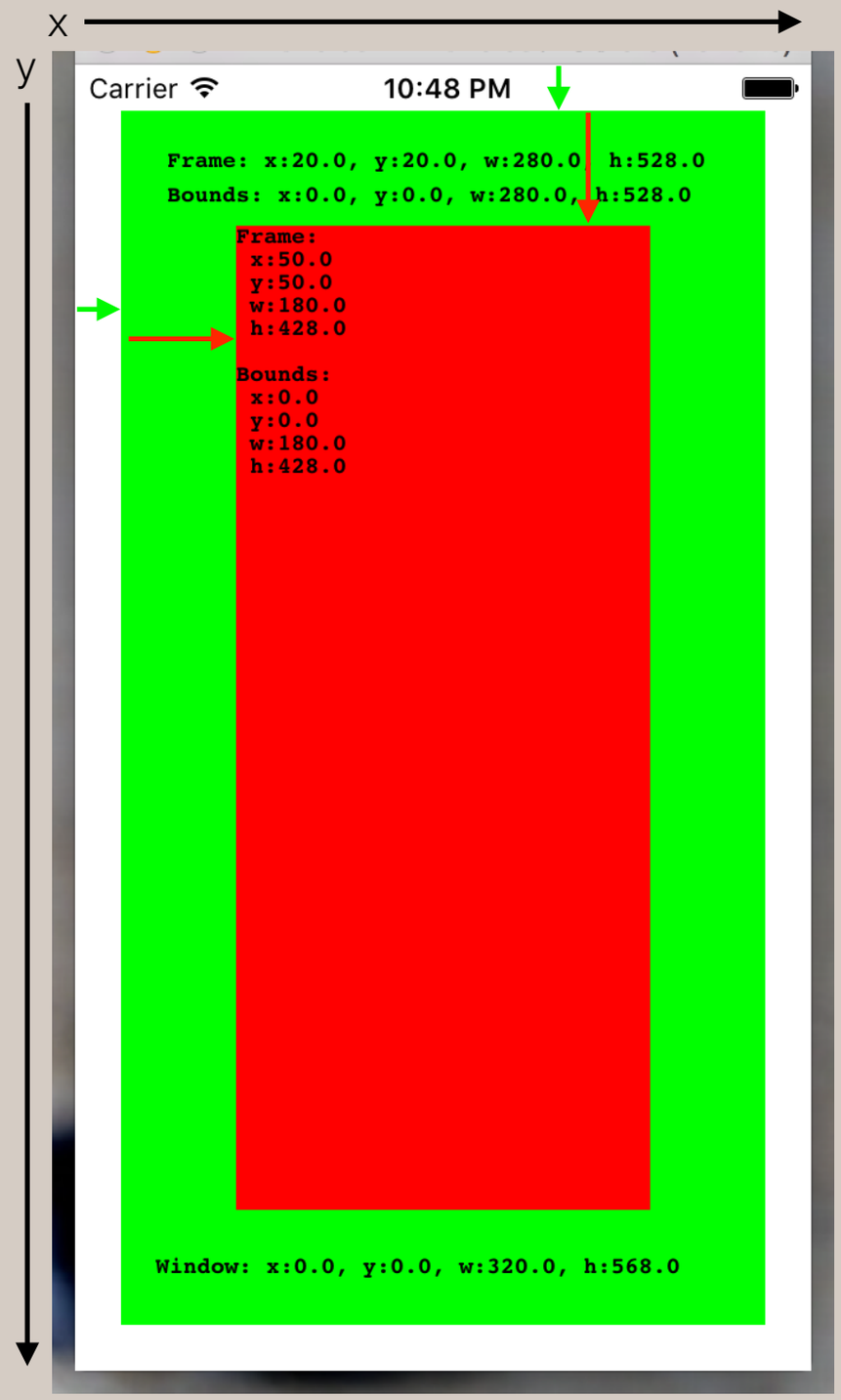
- A view's *frame* is the position of its rectangle in the *super*view's coordinate system. By default it starts at the top left (0, 0)
- A view's *bounds* is a view rectangle in its own coordinate system, typically (0, 0, frameWidth, frameHeight)
- A *center* is a CGPoint expressed in terms of the superview's coordinate system and it determines the position of the exact center point of the view

- In general, these are the relationships:
  - frame.origin = center - (bounds.size / 2.0)
  - center = frame.origin + (bounds.size / 2.0)
  - frame.size = bounds.size

# View Frame and Bounds

Example:

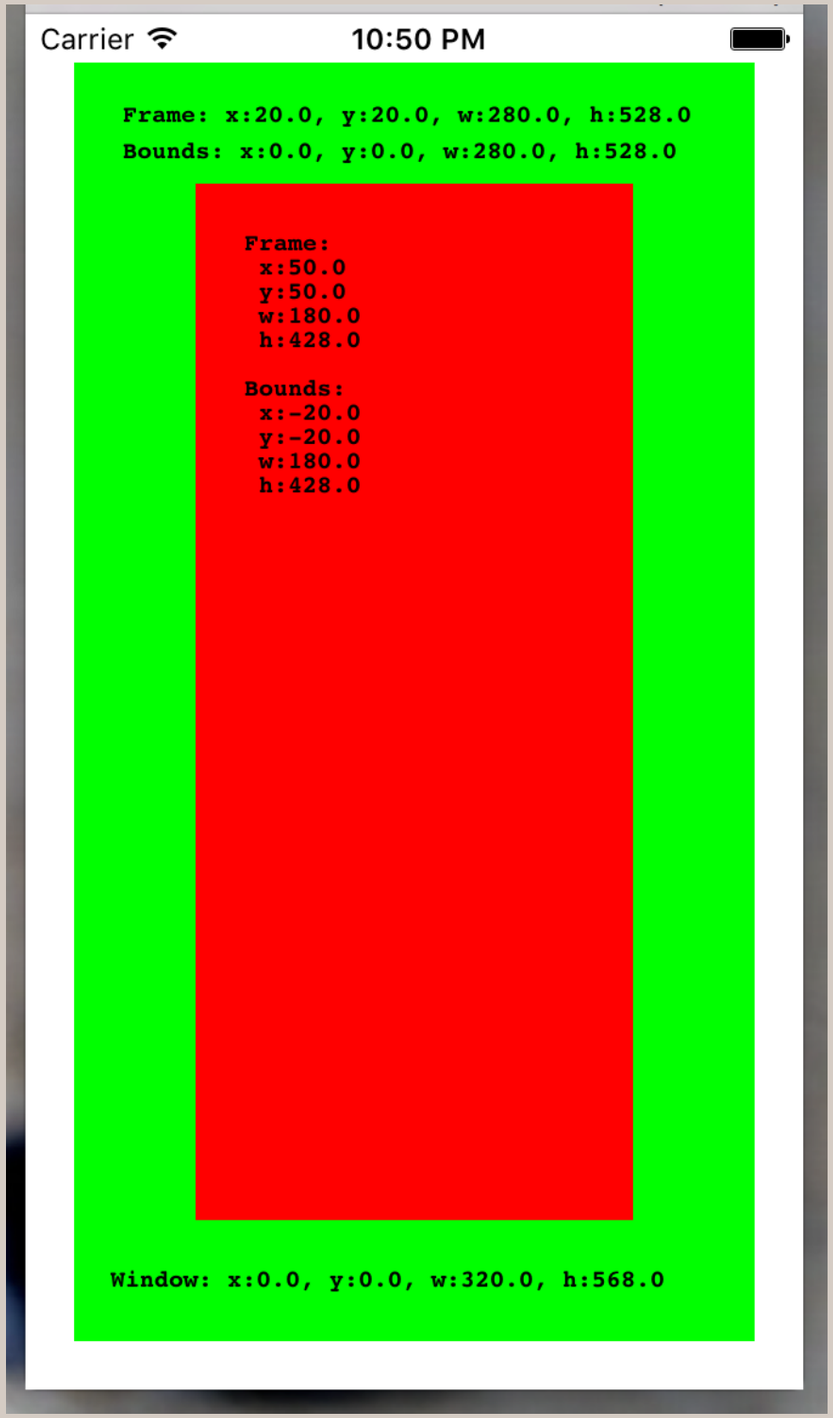
- The white rectangle is the main view
- The green rectangle is a UILabel that is a subview of the main view, offset +20 in the x and y dimension
- The red square is a UILabel that is a subview of the green UILabel, offset +50 in the x and y dimension



# View Frame and Bounds

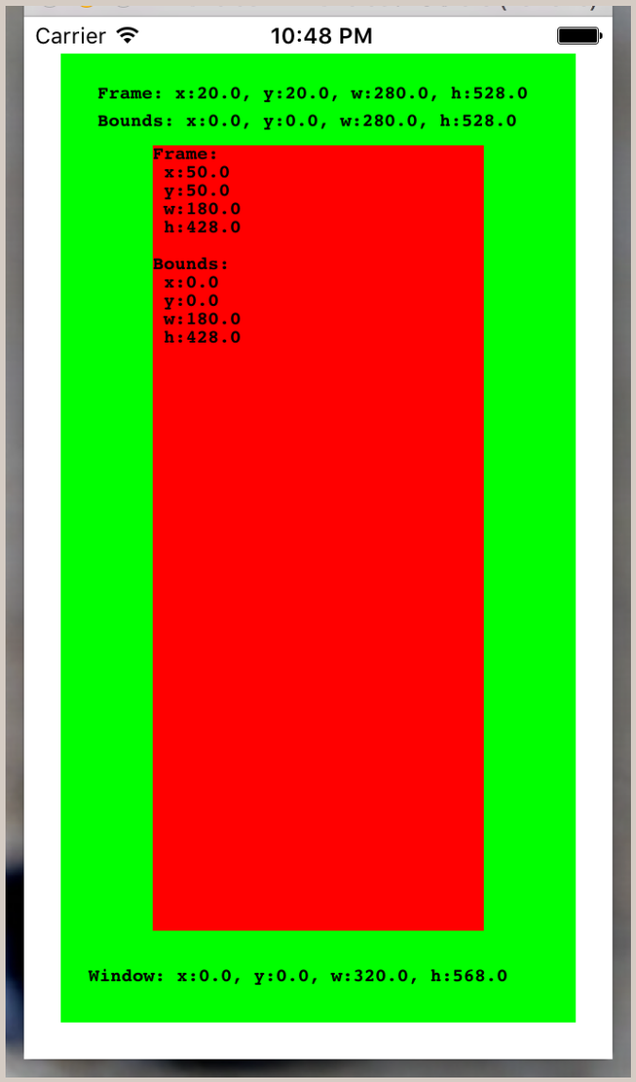
Example:

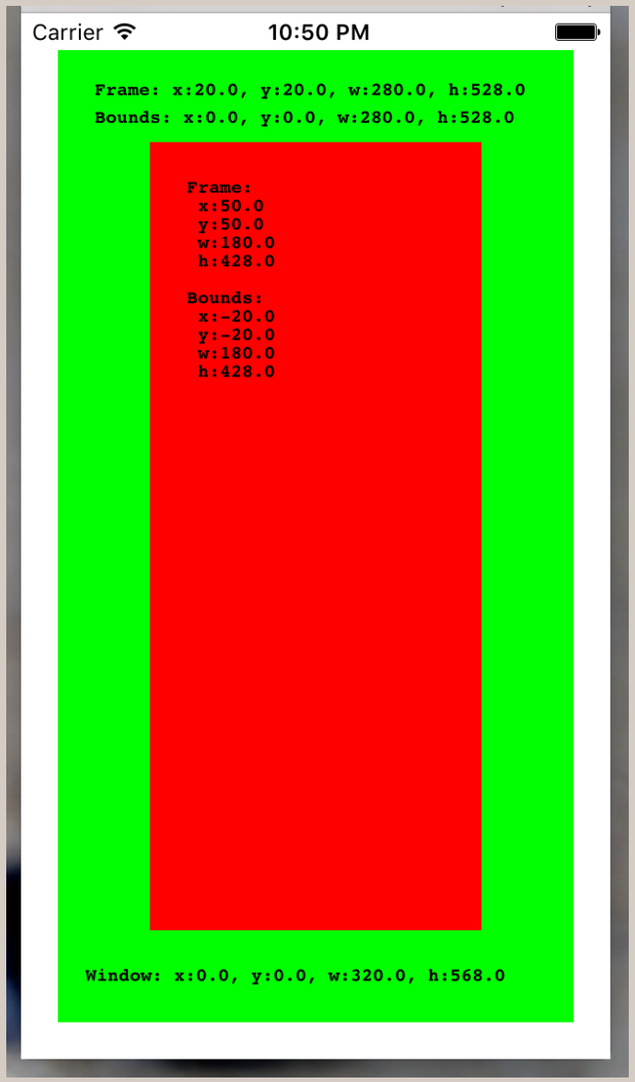
- The red rectangle with bounds x and y set to -20.0
- Notice the indentation, in both the x and y dimensions, of the text in the red box.
- It's because the frame origin for those views remains at (0,0).



# View Frame and Bounds

## Demo: TestFrameAndBoundsSwift



# Other View Properties

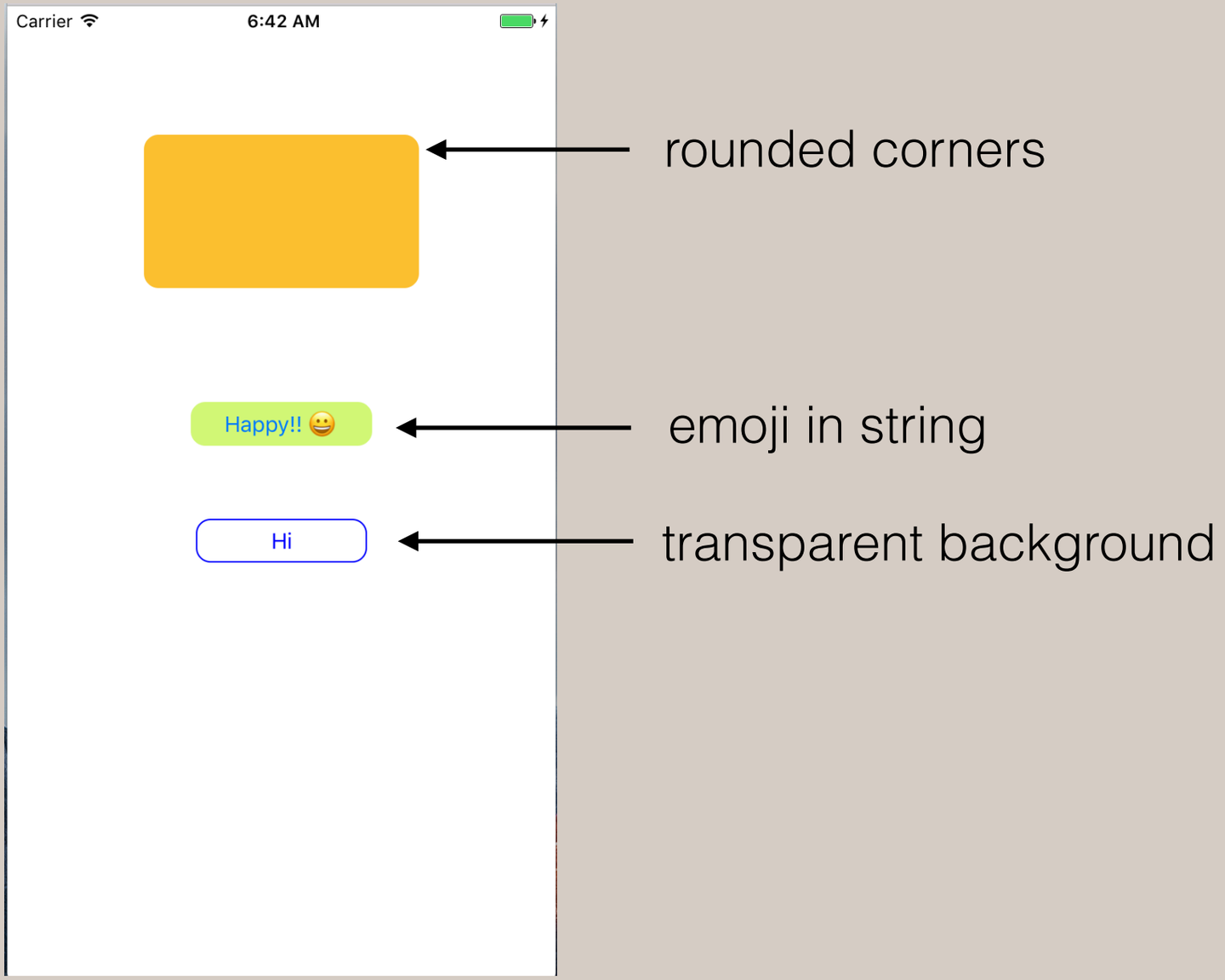
# Other View Properties

Some other UIView properties to manipulate:

- Background color
- Corner radius - to make rounded corners
- Border width
- Border color
- Label text
- Text view text
- Button text
  - Setting some button properties is different because of button states

# Other View Properties

Demo: TestUIViewProperties



# In-Class Exercise

# In-Class Exercise

Create a Single View application that *programmatically* changes properties of some of the UIViews.

Properties to change:

- Background color
- Frame - when tapping button
- Label text
- Button text

# Project

# Project

- Review App Idea Paper document posted to Canvas.
- Meet with your team members.
  - Find out what kind of background each member has.
  - Begin to discuss app ideas.
  - Discuss meeting schedule and set up first meeting.

# Homework 3

# Homework 3

- Define an interface with:
  - A Table View controller.
  - A Navigation controller.
  - A View controller.
- Define constraints for each user interface element.
- Define a segue between the Table View controller and the View controller.
- Pass information from the Table View controller to the View controller, via prepare(for segue).