CTS/ECS 12: Introduction to Media Computation

Welcome!

Contact Info and Background

- Michael Neff
- mpneff@ucdavis.edu
- Interests:
  - Character animation
  - Interactive graphics
  - Computer graphics
  - VR

You…

- How many people have written a computer program before?
- What are your goals for the course?

Course Theme 1

- Learn to program
  - Use Processing (Ben Fry and Casey Reas)
    - Designed for artists
      - Simple and visual
    - Based on Java
      - Not a toy
  - Create Images
  - Manipulate Images and Video
  - Create Interactive Programs
  - Animation
  - Time permitting: 3D and sound

Why learn programming?

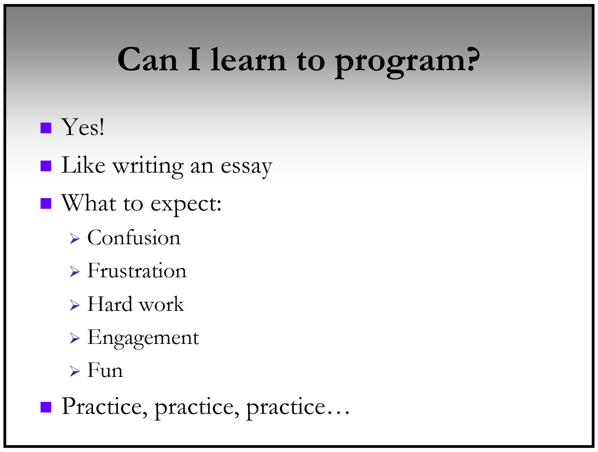
- Control and freedom
- Understand computers
- Fundamental literacy
- Fun!

But I'm a film major!

- Understand how After Effects actually works

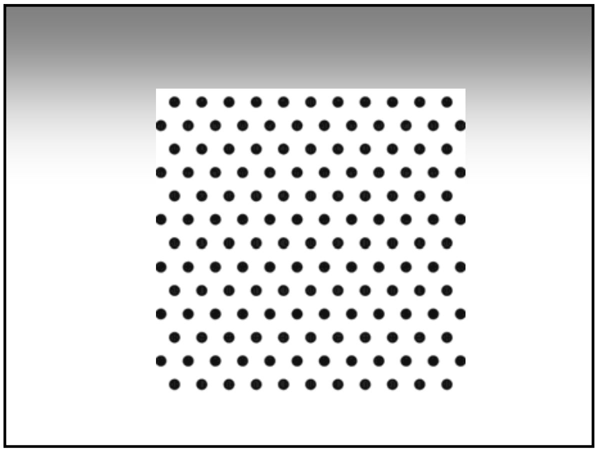
DIGITAL DOMAIN

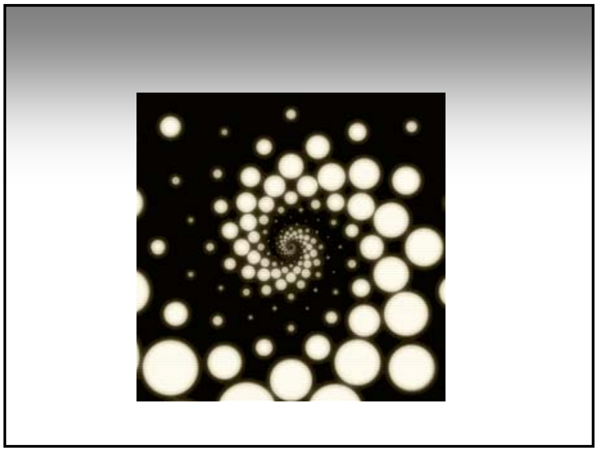
- The media world is changing. What screens are you targeting?

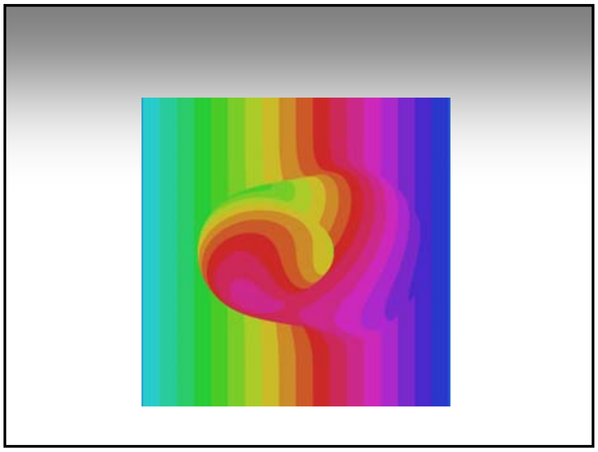


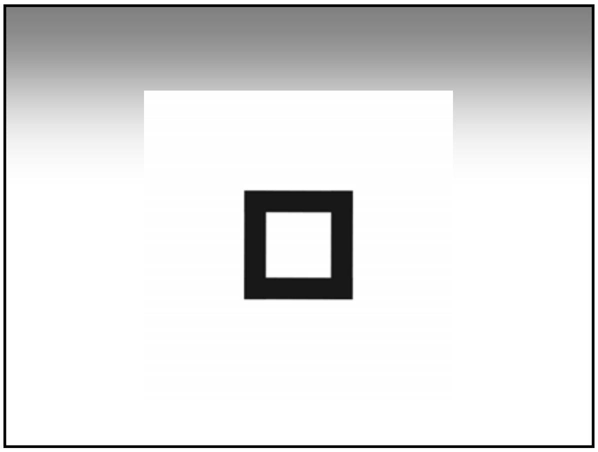
Can I learn to program?

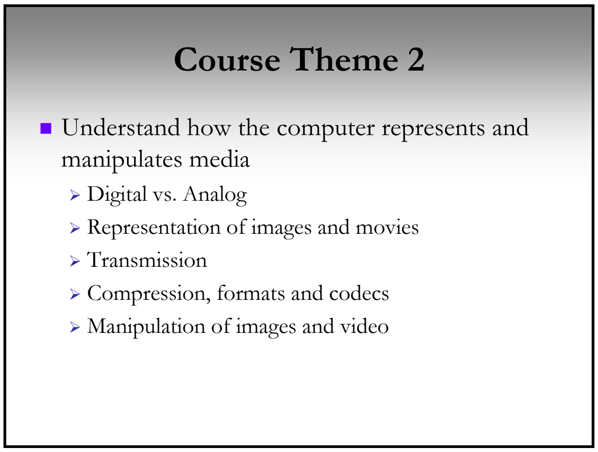
- Yes!
- Like writing an essay
- What to expect:
  - Confusion
  - Frustration
  - Hard work
  - Engagement
  - Fun
- Practice, practice, practice…











Course Theme 2

- Understand how the computer represents and manipulates media
  - Digital vs. Analog
  - Representation of images and movies
  - Transmission
  - Compression, formats and codecs
  - Manipulation of images and video

Support for Your Learning

- Labs
- Drop-in Labs:
  - Wed. 7-9 Hutchison 93
  - Fri. 12-1 Hutchison 73
- Office hours: Tues. 2-3, Thurs. 3-4 Kemper 3031
- Discussion forum on Piazza (link in syllabus and Canvas)
- Online notes
- Course website and Canvas

Labs: 93 Hutchison

- Organized by last name:

Last Name Starts with:	**Your lab:**
A – Do	Mon. 10-11
Du – Huang, W.	Mon. 11-12
Huang, Y – Mar	Mon. 12-1
Mat - Qia	Mon. 3-4
Qua – Thom	Mon. 4-5
Thorn – Z	Mon. 5-6

If you have a *conflict* with your lab time:

- By Thurs., email 12labtime@gmail.com the following:
  - Your name and student number
  - Nature of conflict: (e.g. other class)
  - Lab Assigned
  - Requested Alternative time
  - All times that you can attend
- Start your subject with [12] so they know the email relates to the course

Work in the Course

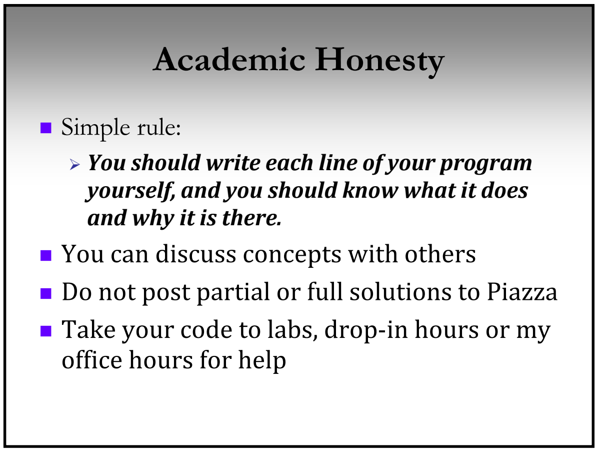
- 6 Assignments
  - Basic Drawing
  - Animation
  - Variables
  - Functions
  - Image Manipulation
  - Interaction Project
- Labs
- Midterm Exam
- Final Exam

Syllabus

On the website. ***Please read it.***

Lecture Capture

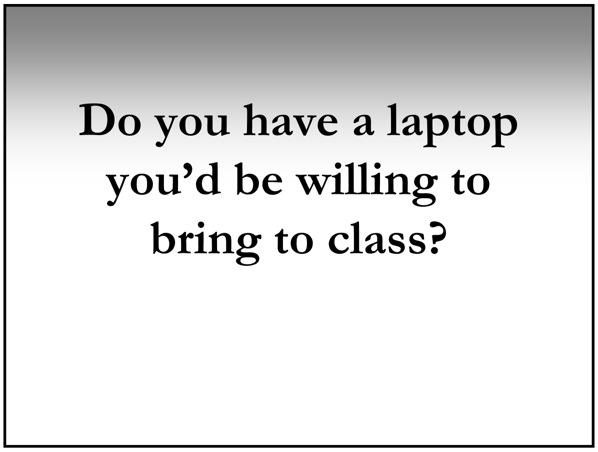
- Lectures will be recorded and posted to Canvas
- How NOT to use Lecture Capture
  - As an excuse to miss class
- How to use Lecture Capture
  - Backup
    - Review something that wasn't clear
    - Refer back to material when studying



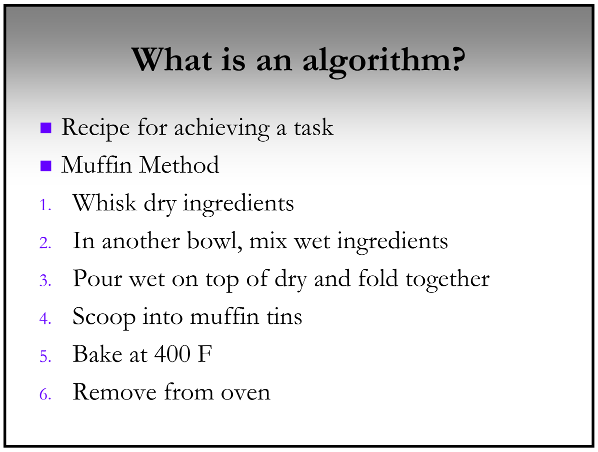
Academic Honesty

- Simple rule:
  - ***You should write each line of your program yourself, and you should know what it does and why it is there.***
- You can discuss concepts with others
- Do not post partial or full solutions to Piazza
- Take your code to labs, drop-in hours or my office hours for help

Back to programming…



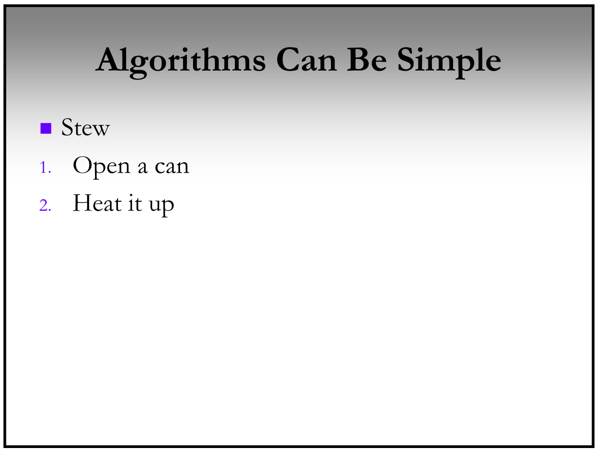
Do you have a laptop you'd be willing to bring to class?



What is an algorithm?

- Recipe for achieving a task
- Muffin Method

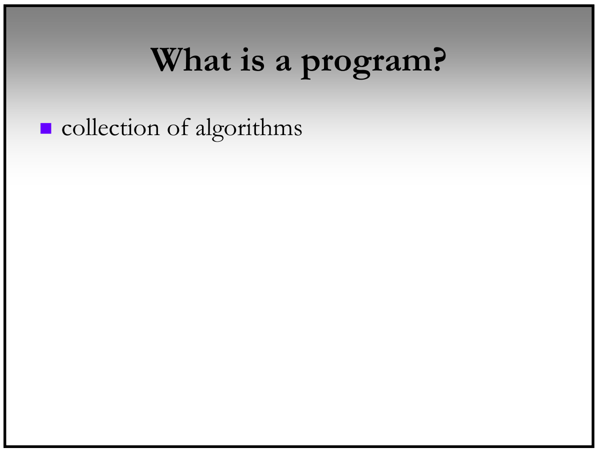
1. Whisk dry ingredients
2. In another bowl, mix wet ingredients
3. Pour wet on top of dry and fold together
4. Scoop into muffin tins
5. Bake at 400 F
6. Remove from oven



Algorithms Can Be Simple

- Stew

1. Open a can
2. Heat it up



What is a program?

- collection of algorithms

How Computer Programs Differ

- *Ridiculously* precise
  - Every detail must be specified
  - Must be written in a very particular way
- Processing helps
  - High level commands

What is programming?

- Writing from scratch
- Modifying
- Extending other people's code (libraries)

Processing

- Go to processing.org
- Download the version for your machine
- Extract the files from the zip file
- Run the "processing" program
- Go with the latest version (3.x)

sketch_130108a | Processing 2.0b7
File Edit Sketch Tools Help
sketch_130108a §

Reading Reference

- Read Chapters 1 and 2 in Shiffman

Example Program

Where do I find commands?

- http://www.processing.org/reference/

Getting Organized

- Group all initialization code

```
void setup()
{
        //all initialization code here  e.g.
        size(400, 400);
}
```

Getting Organized

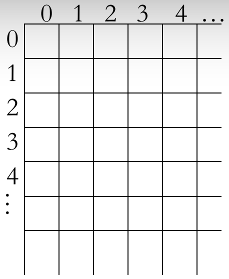
- Group update code (drawing)

```
void draw()
{
        //Anything that changes over time
        stroke(0); //these parameters could be varied
        fill(10);
        rect(360, 200, 40, 200);
}
```

- Called with every update
- Double buffered
  - Update window at end of draw()

Pixels

- Image is a grid of squares called *pixels*
  - Picture element -> pixel
- Like graph paper
  - Filling in the blocks
- (x, y)



A Few Useful Commands

```
size( width, height);//window size
background();//background color
rect(originX,originY,width, height); //create rectangle
ellipse(); //create ellipse
ellipseMode();//ellipse drawing mode
rectMode();//rectangle drawing mode
stroke();//color of line that outlines shape
fill();//color of shape fill
```

Processing Uses States to Modify Commands

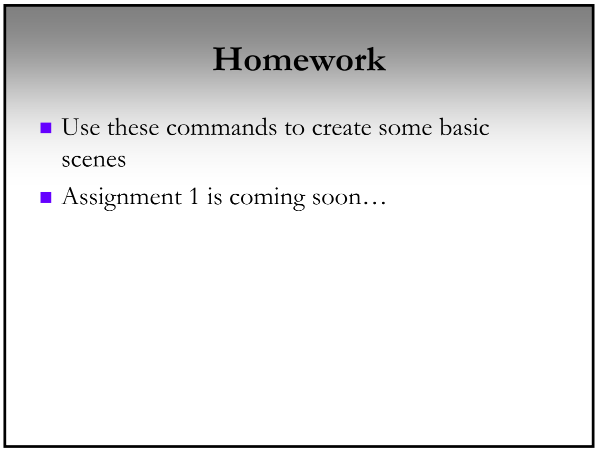
- Set state
- Future commands use the state

```
stroke(0);
fill(150);
rect(200, 200, 40, 200);

fill(10);
rect(360, 200, 40, 200);
```

Processing Uses States to Modify Commands

- State mode used for colors, drawing modes, etc.



Homework

- Use these commands to create some basic scenes
- Assignment 1 is coming soon…