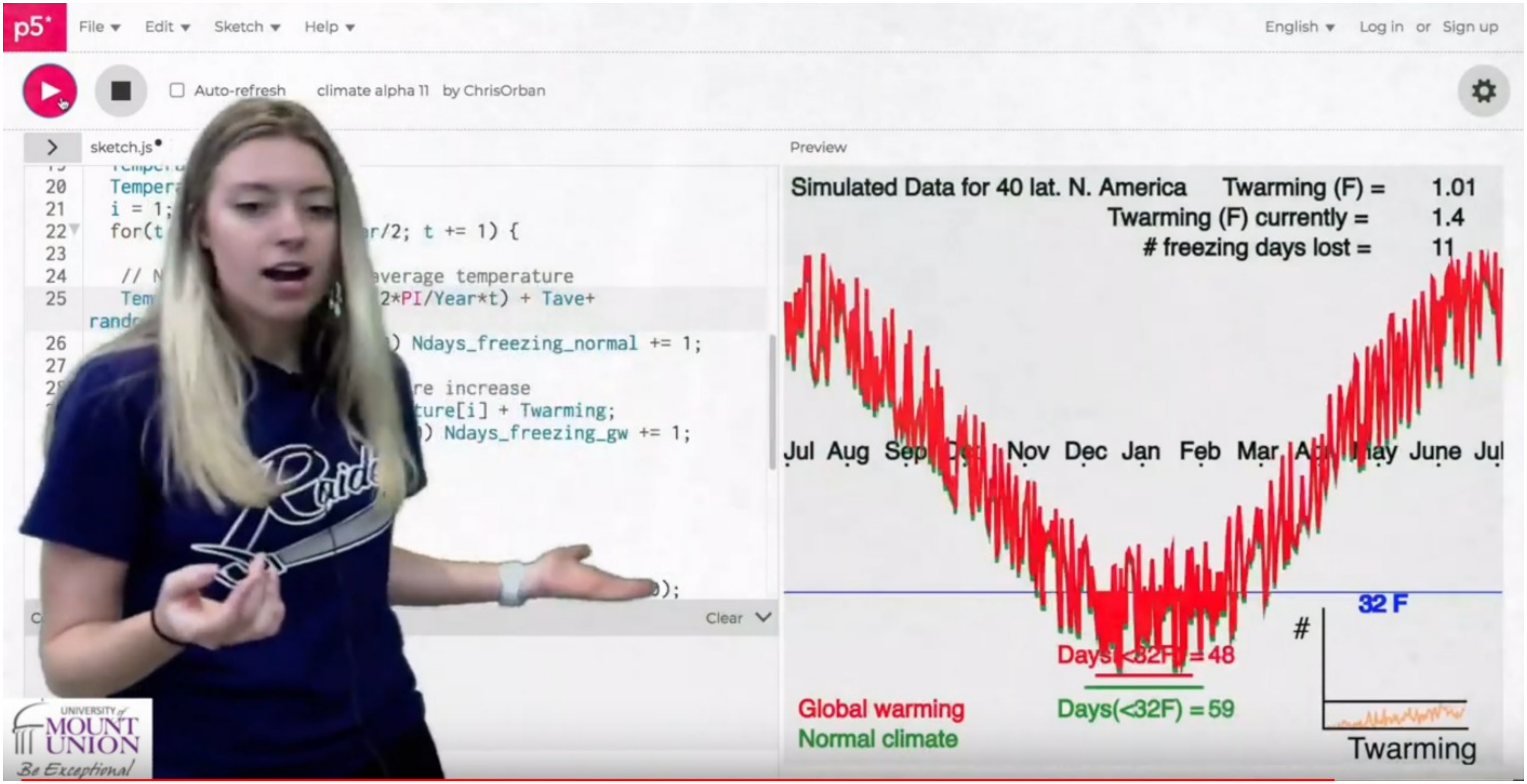
Teaching CS in connection with other subjects



```
  Temper
  i = 1;
  for(t          r/2; t += 1) {

    // N         average temperature
    Tem          2*PI/Year*t) + Tave+
rand
                 ) Ndays_freezing_normal += 1;

                 re increase
                 ture[i] + Twarming;
                 ) Ndays_freezing_gw += 1;

                                            ));
```

```
Simulated Data for 40 lat. N. America     Twarming (F) =        1.01
                              Twarming (F) currently =          1.4
                              # freezing days lost =            11
Days(<32F) = 48
Days(<32F) = 59
```

Prof. Chris Orban

The Ohio State University

STEM coding

A little about me

- Computational physics
- Plasma physics
- Education research
- Ph.D. in Physics from OSU

A little bit about the STEMcoding Project

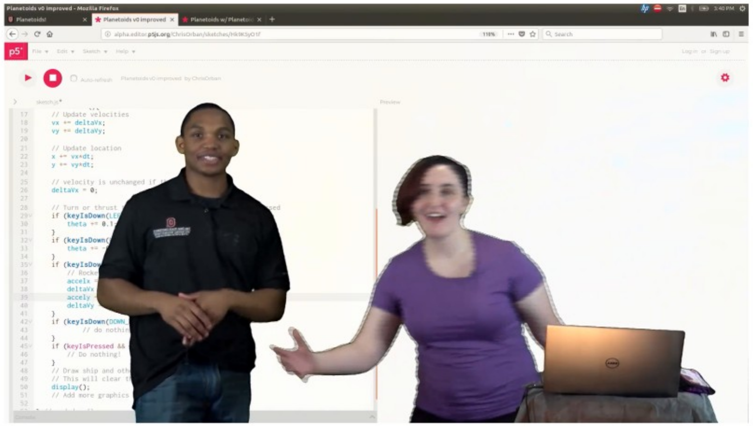
- Started in 2017 with the launch of a youtube channel / hour of code activities
  - Initial focus on physics
- Expanded content to astronomy, physical science, enviro science
- Now connected to a state-wide “discrete math” effort in Ohio*

*Chris Orban / STEMcoding does not speak on behalf of Ohio Dept. of Education and can only share public details

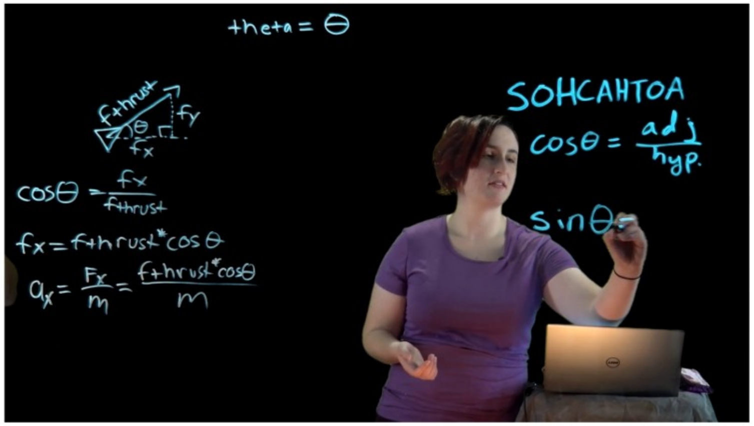
Hour of code activities

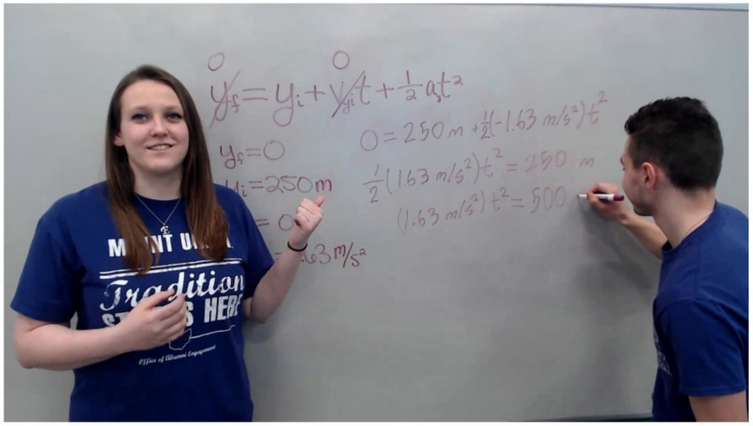
- 2017 release – Move the blob
- 2018 release
  - Asteroids & Lunar Lander
  - Pong & Bonk.io
  - Pi day
- 2019 release
  - Escape Velocity / Newtonian Black Holes
- 2021 release
  - Earth Day / Climate Change activity

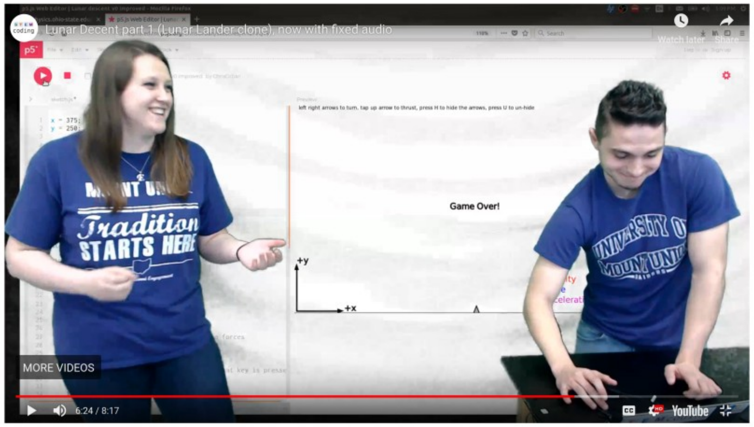
2018 Hour of Code Activities: Asteroids & Lunar Lander


Planetoids v0 improved - Mozilla Firefox
Planetoids!
Planetoids v0 improved
Planetoids w/ Planetoi
alpha.editor.p5js.org/ChrisOrban/sketches/Hk9K5yO1f
p5*
File
Edit
Sketch
Help
Auto-refresh
Planetoids v0 improved by ChrisOrban
sketch.js*
Preview

```
    // Update velocities
    vx += deltaVx;
    vy += deltaVy;

    // Update location
    x += vx*dt;
    y += vy*dt;

    // velocity is unchanged if th
    deltaVx = 0;

    // Turn or thrust                                    sed
    if (keyIsDown(LEF
      theta += 0.1
    }
    if (keyIsDown(
      theta += -
    }
    if (keyIsDow
      // Rocke
      accelx =
      deltaVx
      accely =
      deltaVy
    }
    if (keyIsDown(DOWN_
        // do nothin
    }
    if (keyIsPressed &&
      // Do nothing!
    }
    // Draw ship and oth
    // This will clear t
    display();
    // Add more graphics
```

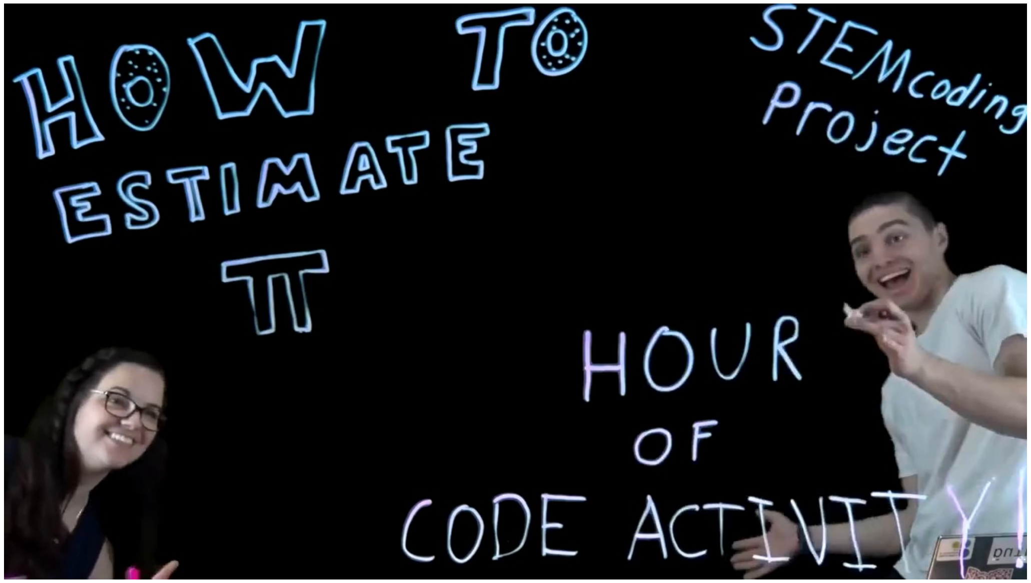

theta = ⊖
f_{thrust}
f_y
f_x
θ
SOHCAHTOA
$\cos\theta = \frac{adj}{hyp.}$
$\sin\theta =$
$\cos\theta = \frac{f_x}{f_{thrust}}$
$f_x = f_{thrust} * \cos\theta$
$a_x = \frac{F_x}{m} = \frac{f_{thrust} * \cos\theta}{m}$

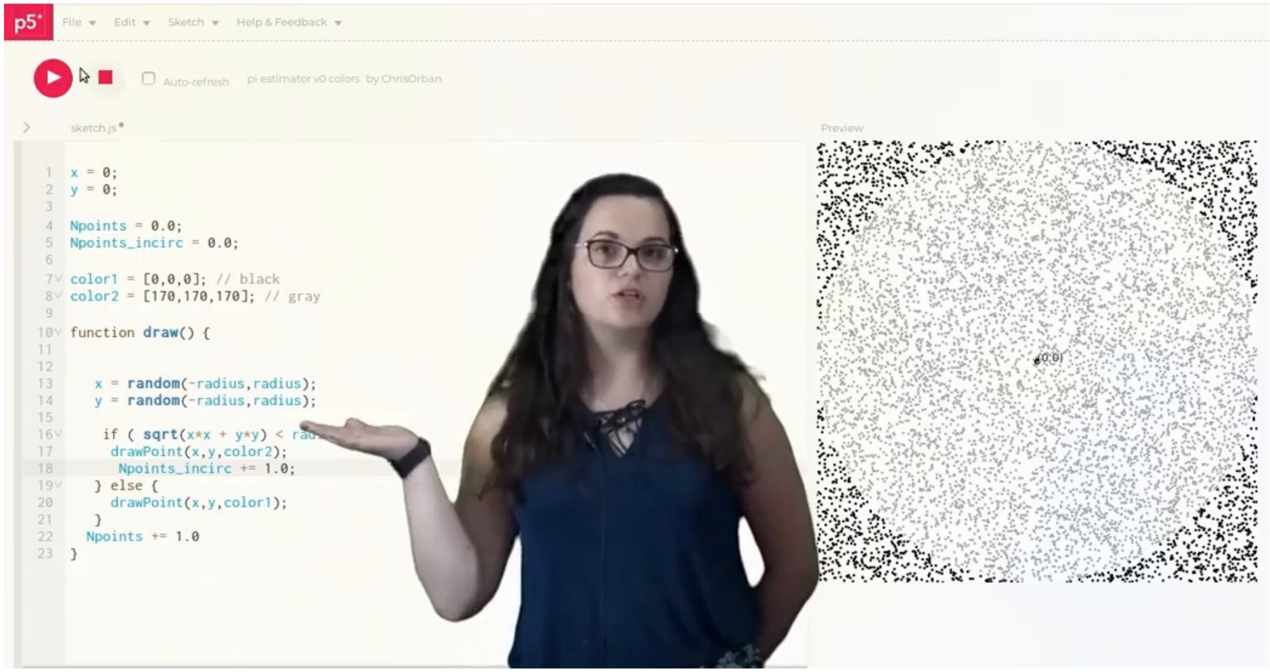

$\cancel{y_f} = y_i + \cancel{v_{yi}} t + \frac{1}{2} a t^2$
$0 = 250\,m + \frac{1}{2}(-1.63\,m/s^2) t^2$
$\frac{1}{2}(1.63\,m/s^2) t^2 = 250\,m$
$(1.63\,m/s^2) t^2 = 500$
$y_f = 0$
$y_i = 250\,m$
$= 0$
$= -1.63\,m/s^2$
Mount Union
Tradition Starts Here


Lunar Decent part 1 (Lunar Lander clone), now with fixed audio
Watch later
Share
p5*
left right arrows to turn, tap up arrow to thrust, press H to hide the arrows, press U to un-hide
Game Over!
+y
+x
Mount Union
Tradition Starts Here
University of Mount Union
MORE VIDEOS
6:24 / 8:17
YouTube

STEMcoding

2018 Hour of Code Activities: Pi day

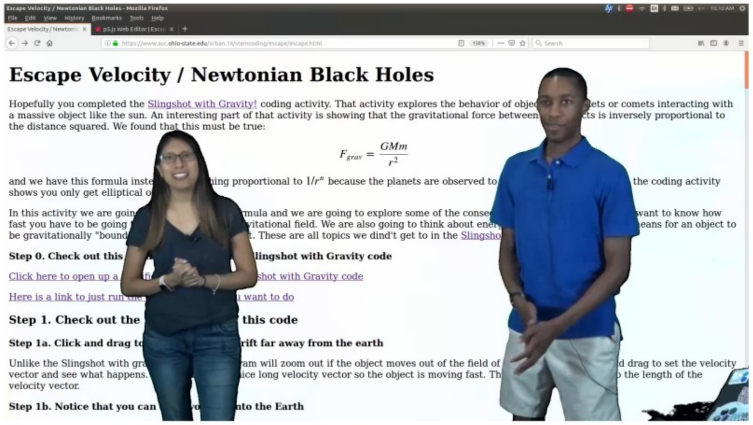




```
x = 0;
y = 0;

Npoints = 0.0;
Npoints_incirc = 0.0;

color1 = [0,0,0]; // black
color2 = [170,170,170]; // gray

function draw() {


   x = random(-radius,radius);
   y = random(-radius,radius);

    if ( sqrt(x*x + y*y) < rad
     drawPoint(x,y,color2);
      Npoints_incirc += 1.0;
   } else {
     drawPoint(x,y,color1);
   }
  Npoints += 1.0
}
```

STEMcoding

2019 Hour of Code Activities: Escape Velocity & Black Holes


Escape Velocity / Newtonian Black Holes - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Escape Velocity / Newtonia
p5.js Web Editor | Escap
https://www.asc.ohio-state.edu/orban.14/stemcoding/escape/escape.html
150%
Search
10:10 AM

Escape Velocity / Newtonian Black Holes

Hopefully you completed the Slingshot with Gravity! coding activity. That activity explores the behavior of obje lets or comets interacting with a massive object like the sun. An interesting part of that activity is showing that the gravitational force between cts is inversely proportional to the distance squared. We found that this must be true:

$$F_{grav} = \frac{GMm}{r^2}$$

and we have this formula inste hing proportional to $1/r^n$ because the planets are observed to the coding activity shows you only get elliptical o

In this activity we are goin rmula and we are going to explore some of the conse want to know how fast you have to be going vitational field. We are also going to think about ener eans for an object to be gravitationally "bound t. These are all topics we dind't get to in the Slingsho

Step 0. Check out this lingshot with Gravity code

Click here to open up a fi shot with Gravity code

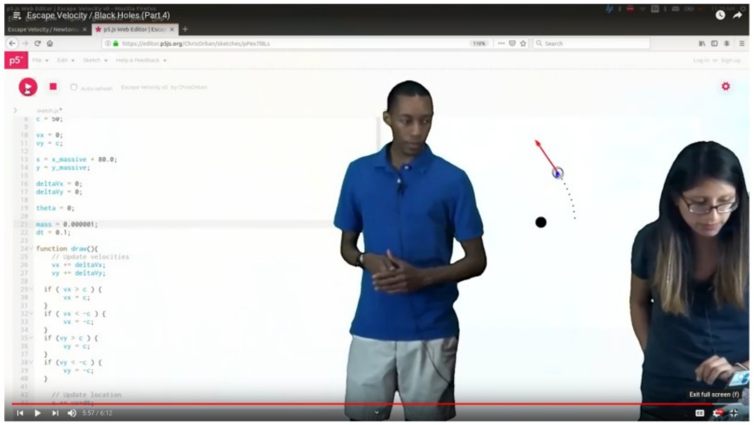
Here is a link to just run the u want to do

Step 1. Check out the this code

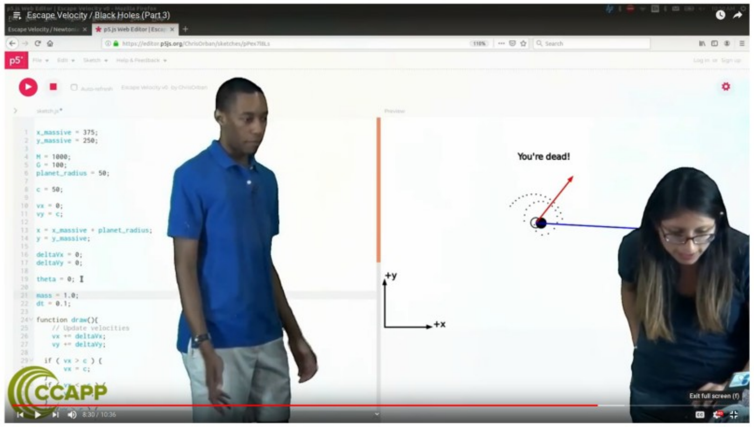
Step 1a. Click and drag to rift far away from the earth

Unlike the Slingshot with gra am will zoom out if the object moves out of the field of d drag to set the velocity vector and see what happens. ice long velocity vector so the object is moving fast. Th the length of the velocity vector.

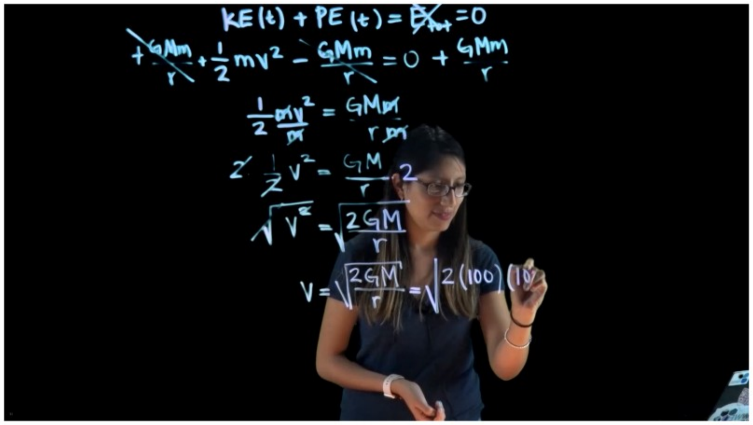
Step 1b. Notice that you can vo nto the Earth


Escape Velocity / Black Holes (Part 4)
Escape Velocity / Newtonia
p5.js Web Editor | Escap
https://editor.p5js.org/ChrisOrban/sketches/pPex7l8Ls
110%
Search
p5*
File
Edit
Sketch
Help & Feedback
Log in or Sign up
Auto-refresh
Escape Velocity v0 by ChrisOrban
sketch.js*
Exit full screen (f)
5:57 / 6:12

```
c = 50;

vx = 0;
vy = c;

x = x_massive + 80.0;
y = y_massive;

deltaVx = 0;
deltaVy = 0;

theta = 0;

mass = 0.000001;
dt = 0.1;

function draw(){
    // Update velocities
    vx += deltaVx;
    vy += deltaVy;

  if ( vx > c ) {
      vx = c;
  }
  if ( vx < -c ) {
      vx = -c;
  }
  if (vy > c ) {
      vy = c;
  }
  if (vy < -c ) {
      vy = -c;
  }

    // Update location
```



```
x_massive = 375;
y_massive = 250;

M = 1000;
G = 100;
planet_radius = 50;

c = 50;

vx = 0;
vy = c;

x = x_massive + planet_radius;
y = y_massive;

deltaVx = 0;
deltaVy = 0;

theta = 0;

mass = 1.0;
dt = 0.1;

function draw(){
    // Update velocities
    vx += deltaVx;
    vy += deltaVy;

  if ( vx > c ) {
      vx = c;
```



$$KE(t) + PE(t) = \cancel{E_{tot}} = 0$$

$$\cancel{+\frac{GMm}{r}} + \frac{1}{2}mv^2 - \cancel{\frac{GMm}{r}} = 0 + \frac{GMm}{r}$$

$$\frac{1}{2}\frac{\cancel{m}v^2}{\cancel{m}} = \frac{GM\cancel{m}}{r\cancel{m}}$$

$$\cancel{2}\,\frac{1}{\cancel{2}}v^2 = \frac{GM}{r}\,2$$

$$\sqrt{v^2} = \sqrt{\frac{2GM}{r}}$$

$$v = \sqrt{\frac{2GM}{r}} = \sqrt{2(100)(10}$$

S T E M coding

2021 Hour of Code Activity: Earth Day

Data Science: Earth Day

Introduction

Step 1. Get Climate Data

Step 2. Calculate Average Temperatures

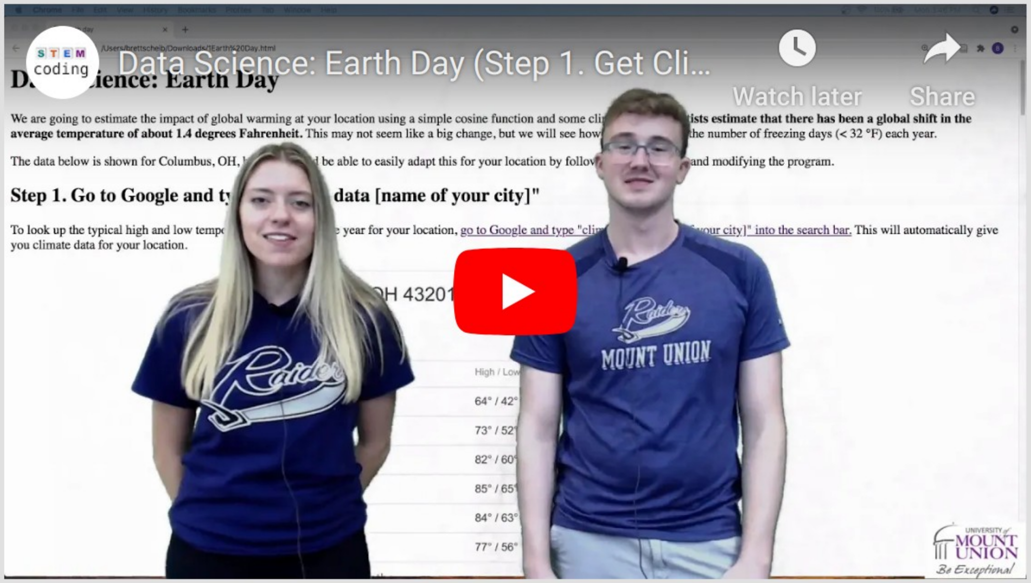
Step 3. Open the Code

Step 4. Add Your Temperatures

Step 5. Run the Code

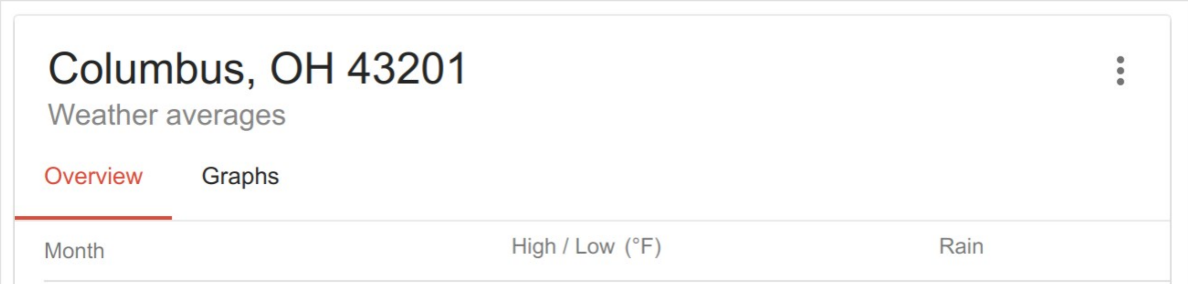
Step 6. Temperature Variability

Step 1. Go to Google and collect climate data



To look up the typical high and low temperatures throughout the year for your location, go to Google and type "climate data [name of your city]" into the search bar. This will automatically give you climate data for your location.

The data below is shown for Columbus, OH, but you should be able to easily adapt this for your location by following the directions and modifying the program.



Columbus, OH 43201

Weather averages

Overview Graphs

Month	High / Low (°F)	Rain

Thoughts on Connectedness

Katie Bouman!

Connectedness and standards

- “Computational thinking” (CT) became popularized by an essay in 2006
- In 2013, the Next Generation Science Standards (NGSS) mentions “computational thinking” as a tool for doing science
- Data science (DS) has gotten more attention in the last few years
- DS and CT are interconnected in many ways but in other ways the communities are disconnected

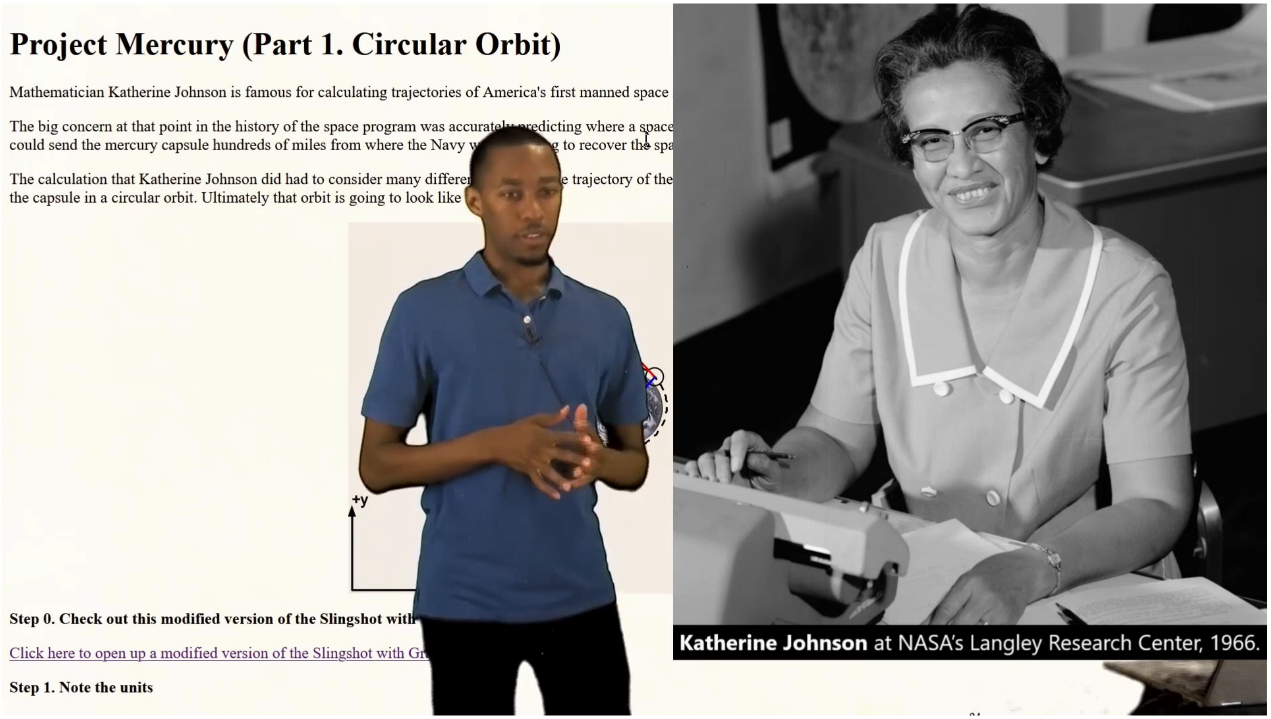
Final thoughts

1. Math and science teachers were doing data science and computational thinking before it was cool

2. The need for more data science / data skills is DIRECTLY connected to long-time struggles to get kids to understand graphs and plots

How would I connect with...

Earth Science / Astronomy



Project Mercury (Part 1. Circular Orbit)

Mathematician Katherine Johnson is famous for calculating trajectories of America's first manned space

The big concern at that point in the history of the space program was accurately predicting where a space could send the mercury capsule hundreds of miles from where the Navy w g to recover the spa

The calculation that Katherine Johnson did had to consider many differer e trajectory of the the capsule in a circular orbit. Ultimately that orbit is going to look like

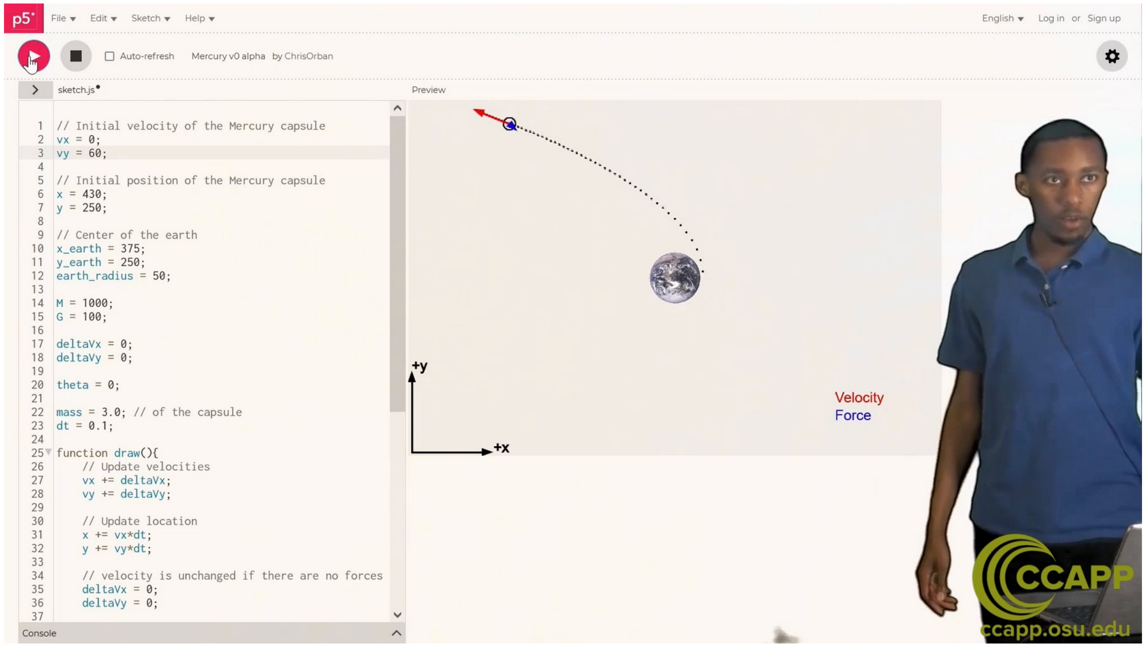
Step 0. Check out this modified version of the Slingshot with

Click here to open up a modified version of the Slingshot with Gr

Step 1. Note the units

Katherine Johnson at NASA's Langley Research Center, 1966.

Project Mercury Activity



```
// Initial velocity of the Mercury capsule
vx = 0;
vy = 60;

// Initial position of the Mercury capsule
x = 430;
y = 250;

// Center of the earth
x_earth = 375;
y_earth = 250;
earth_radius = 50;

M = 1000;
G = 100;

deltaVx = 0;
deltaVy = 0;

theta = 0;

mass = 3.0; // of the capsule
dt = 0.1;

function draw(){
    // Update velocities
    vx += deltaVx;
    vy += deltaVy;

    // Update location
    x += vx*dt;
    y += vy*dt;

    // velocity is unchanged if there are no forces
    deltaVx = 0;
    deltaVy = 0;

```

STEM coding

STEMcoding Astronomy Youtube Playlist

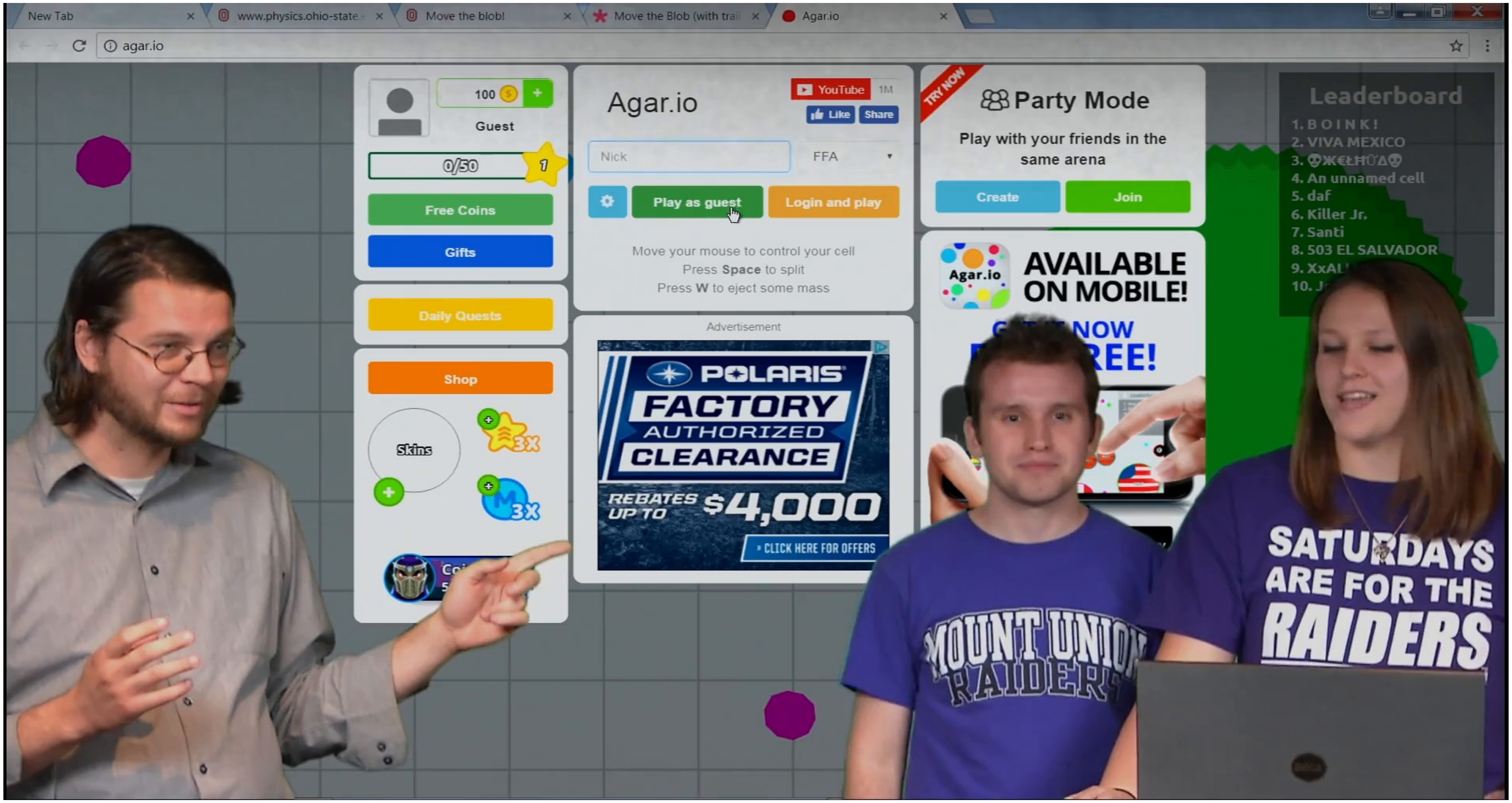
Jimmy Newland's Website

(www.jimmynewland.com)

How would I connect with...

Physical Science

Move the blob



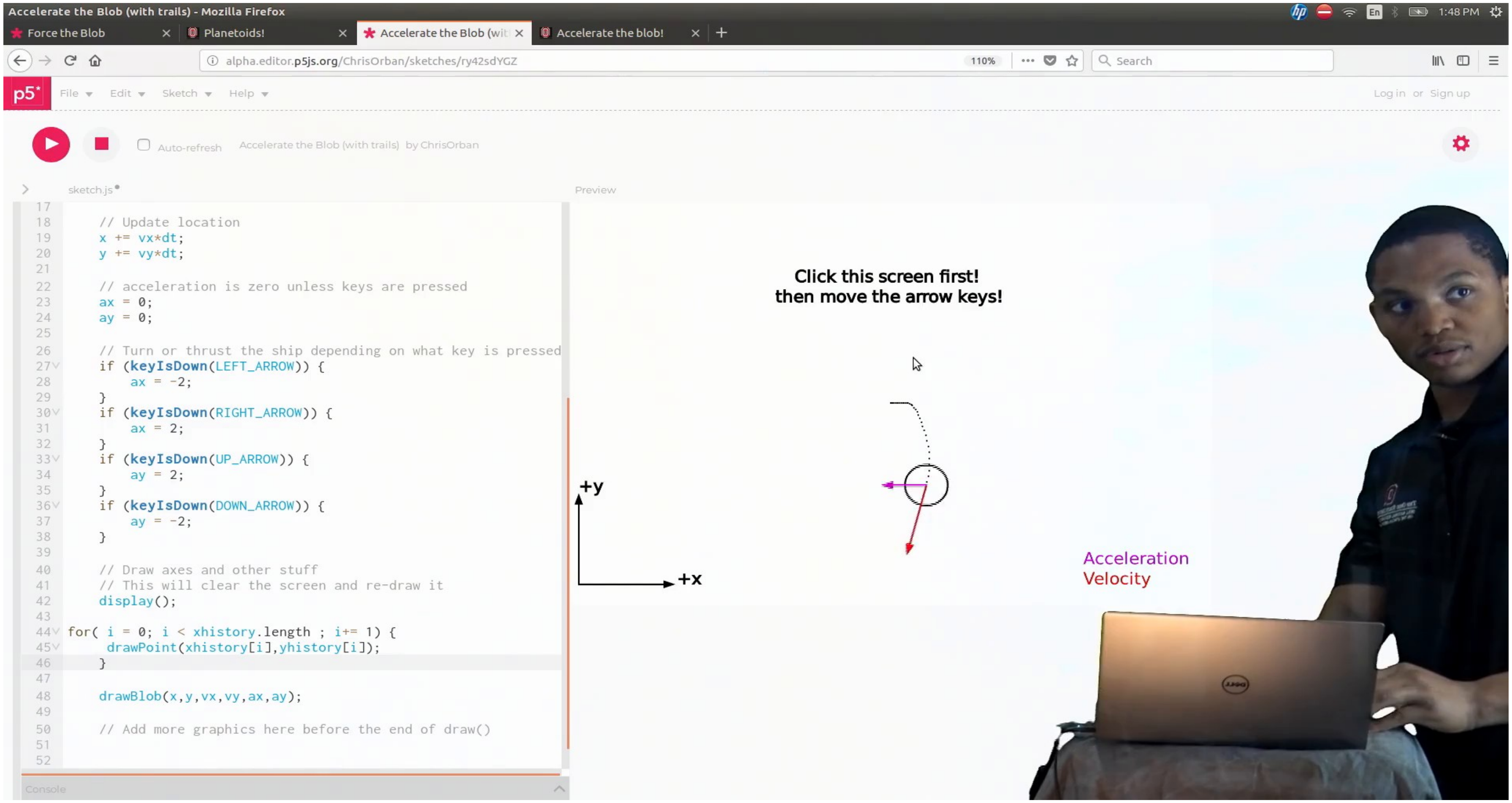
Play with your friends in the same arena

Move your mouse to control your cell
Press **Space** to split
Press **W** to eject some mass

http://go.osu.edu/movetheblob2

STEMcoding

Accelerate the blob



```
    // Update location
    x += vx*dt;
    y += vy*dt;

    // acceleration is zero unless keys are pressed
    ax = 0;
    ay = 0;

    // Turn or thrust the ship depending on what key is pressed
    if (keyIsDown(LEFT_ARROW)) {
        ax = -2;
    }
    if (keyIsDown(RIGHT_ARROW)) {
        ax = 2;
    }
    if (keyIsDown(UP_ARROW)) {
        ay = 2;
    }
    if (keyIsDown(DOWN_ARROW)) {
        ay = -2;
    }

    // Draw axes and other stuff
    // This will clear the screen and re-draw it
    display();

for( i = 0; i < xhistory.length ; i+= 1) {
     drawPoint(xhistory[i],yhistory[i]);
    }

    drawBlob(x,y,vx,vy,ax,ay);

    // Add more graphics here before the end of draw()
```

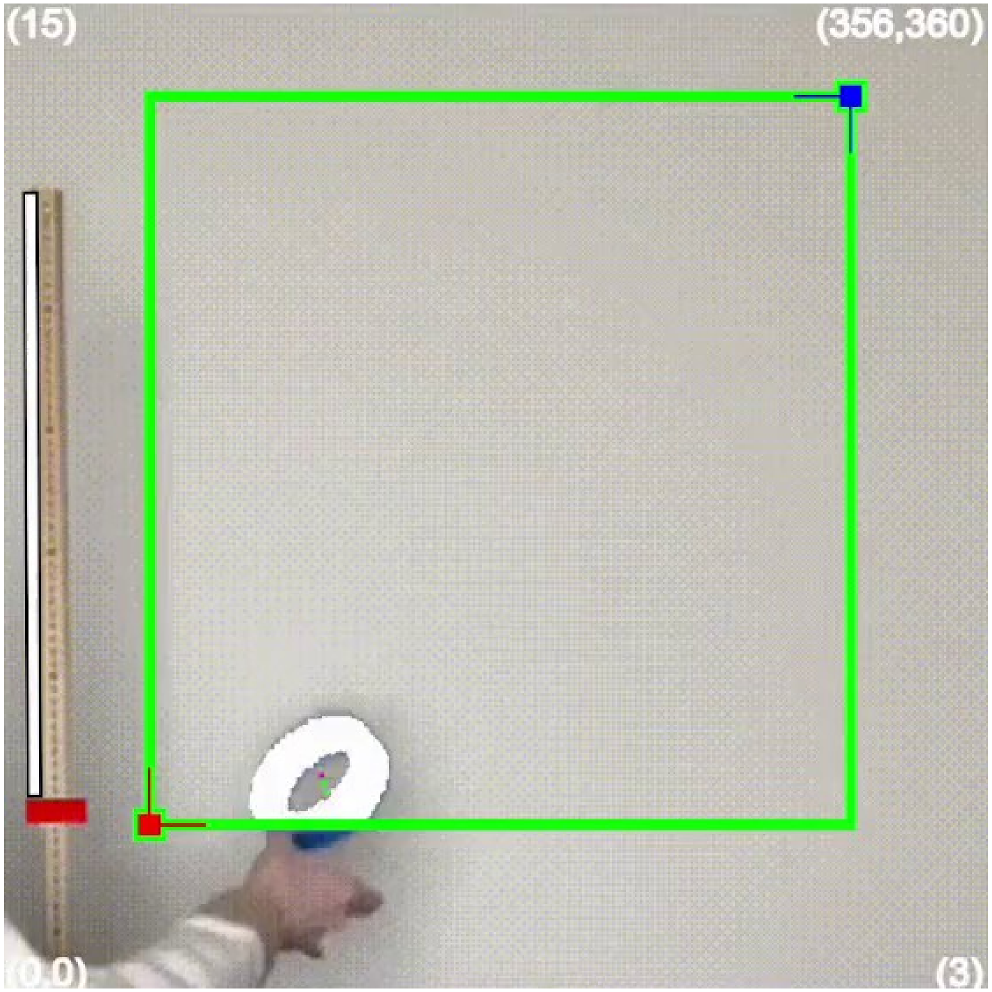
Click this screen first!
then move the arrow keys!

http://go.osu.edu/acceleratetheblob2

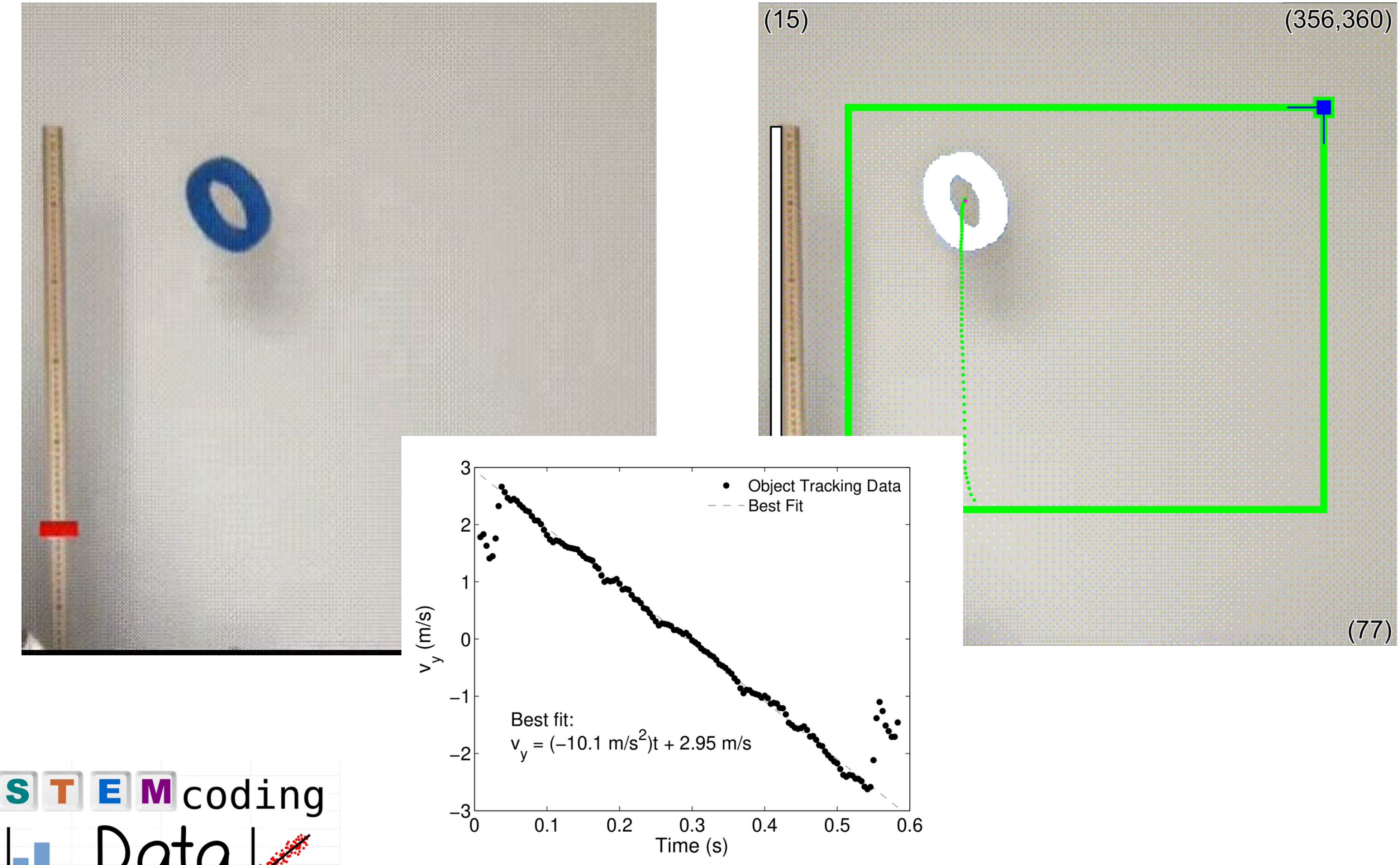
STEM coding

STEMcoding Object Tracker

Track colors not objects!



Track colors not objects!



STEM coding

Data Science

STEMcoding Data Science

arXiv:2206.07909

Methods to Simplify Object Tracking in Video Data

Orban, C. M.,[1,2] Zimmerman, S.,[3,4,5] Kulp, J.,[1] Boughton, J.,[6] Perrico, Z.,[7] Rapp, B.,[7] and Teeling-Smith, R.[7]

[1] *Physics Department, The Ohio State University, 191 W Woodruff Ave, Columbus, OH 43210*
[2] *Physics Department, The Ohio State University at Marion, 1461 Mount Vernon Ave, Marion, OH 43302*
[3] *Department of Mathematics, The Ohio State University, 231 West 18th Avenue, Columbus, OH 43210*
[4] *Department of Mathematics, The Ohio State University at Marion, 1461 Mount Vernon Ave, Marion, OH 43302*
[5] *Mathematics Department, The Ohio State University at Marion, 1461 Mount Vernon Ave, Marion, OH 43302*
[6] *Great Oaks Career Campuses, 110 Great Oaks Drive, Cincinnati, OH, 45241*
[7] *University of Mount Union, Alliance, OH, 44601*

I. INTRODUCTION

Recent years have seen an explosion of interest in analyzing the motion of objects in video data as a way for students to connect the concepts of physics to something tangible like a video recording of an experiment [1, 2]. The limitations of distance learning during the COVID-19 pandemic especially grew interest in this area because students could not attend in person lab activities, but they could analyze video data from their computers at home, which is what many instructors chose to focus on.

Generally, the goal of a student activity involving analysis of video data is to obtain the x,y position of a particular object in as many frames of the video as possible. Once obtained, this data can be used to infer velocities, acceleration and any number of other quantities like momentum or energy. A variety of software exists for students to look at individual frames and click on the object to infer the x,y position (e.g. [2–6]). For longer videos or videos recorded at a high frame rate this task can become tedious. Some, but not all, of these tools include a capability to automatically identify the position of the object in the frame [2–4]. But it is not unusual, especially when inexperienced users are recording the video and config-

Video to analyze a video where he is walking across the sidewalk [2]. In the analyzed video, Dr. Lane has a red piece of paper taped to his right shoulder. This piece of paper is the object being tracked.

In the video tutorial, Dr. Lane explains that it is important to define the “template image" of the object not at the center of the paper but rather at one of the corners so that the object tracking algorithm searches for a splotch of red next to some gray, which is the color of his shirt. With this hint, the object tracking works well and the plot of x versus time shows that Dr. Lane is walking at an approximately constant velocity as expected.

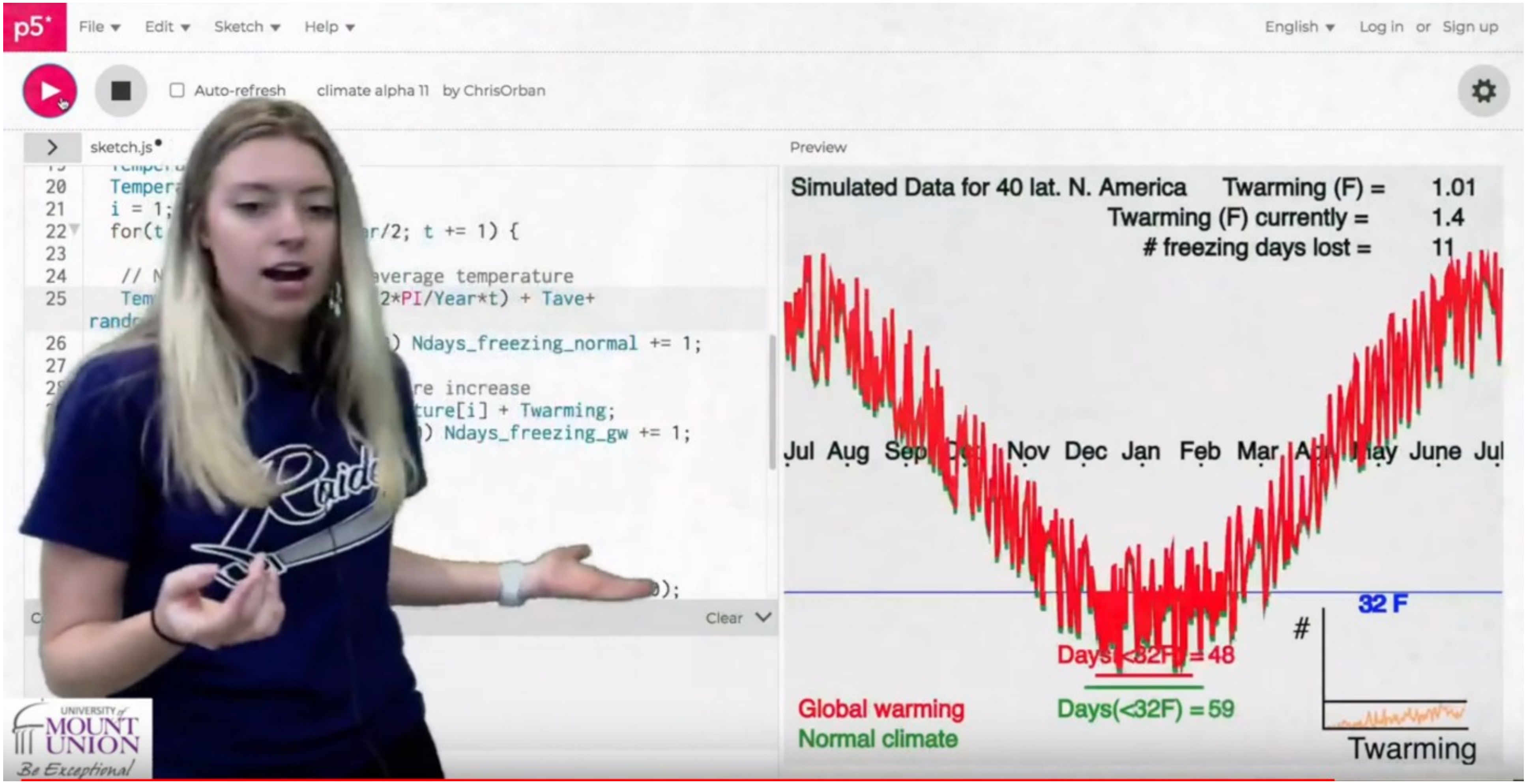
Although this is just a brief moment in the tutorial video it underscores the difficulty of what automatic object tracking is attempting to do. The background of Fig. 1, for example, has many different features that the program potentially needs to scan through as it searches for the the red piece of paper. Further complicating matters, the paper itself will look different in each frame due to changes in lighting and shadows. In our own experimentation with object tracking using a different program – specifically the Vernier Video Physics app – we found it to be surprisingly difficult to get the program to “lock on" to a moving object.

For the full list of physsci, physics, & astro activities:

http://go.osu.edu/physics_coding

How would I connect with...

Earth Science / Environmental
Science / Biology /
Atmospheric Science



```
  Temper
  i = 1;
  for(t                    r/2; t += 1) {

    // N                   verage temperature
    Tem                    2*PI/Year*t) + Tave+
rand
                          ) Ndays_freezing_normal += 1;

                           re increase
                           ture[i] + Twarming;
                          ) Ndays_freezing_gw += 1;
```

Simulated Data for 40 lat. N. America Twarming (F) = 1.01
Twarming (F) currently = 1.4
# freezing days lost = 11

Days(<32F) = 48

Days(<32F) = 59

Global warming

Normal climate

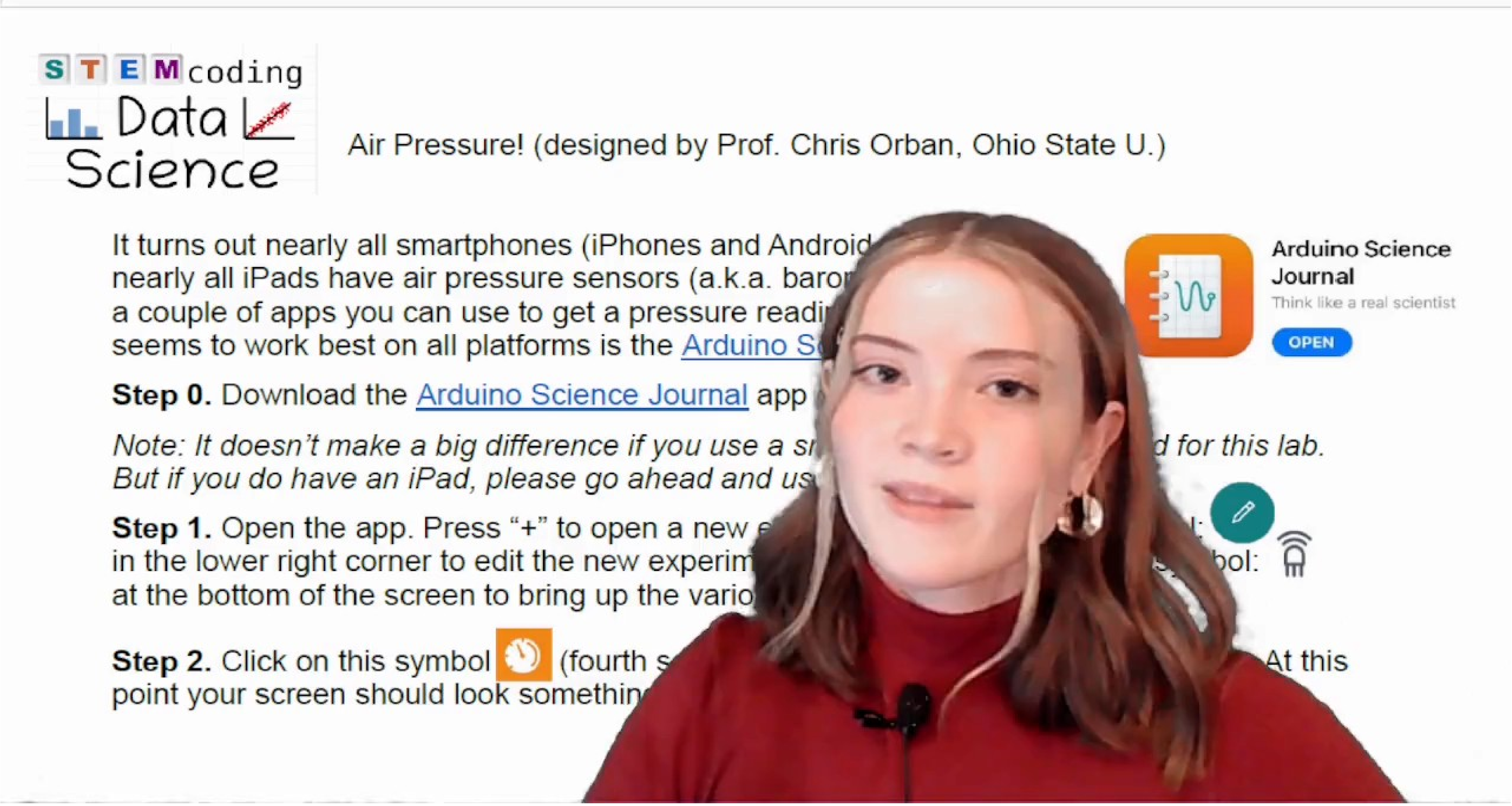
http://go.osu.edu/earthdaycoding

STEM coding

Data Science

How would I connect with...

Earth Science / Atmospheric
Science



Air Pressure! (designed by Prof. Chris Orban, Ohio State U.)

It turns out nearly all smartphones (iPhones and Android
nearly all iPads have air pressure sensors (a.k.a. baro
a couple of apps you can use to get a pressure readi
seems to work best on all platforms is the Arduino Sc

Step 0. Download the Arduino Science Journal app

*Note: It doesn't make a big difference if you use a sı d for this lab.
But if you do have an iPad, please go ahead and us*

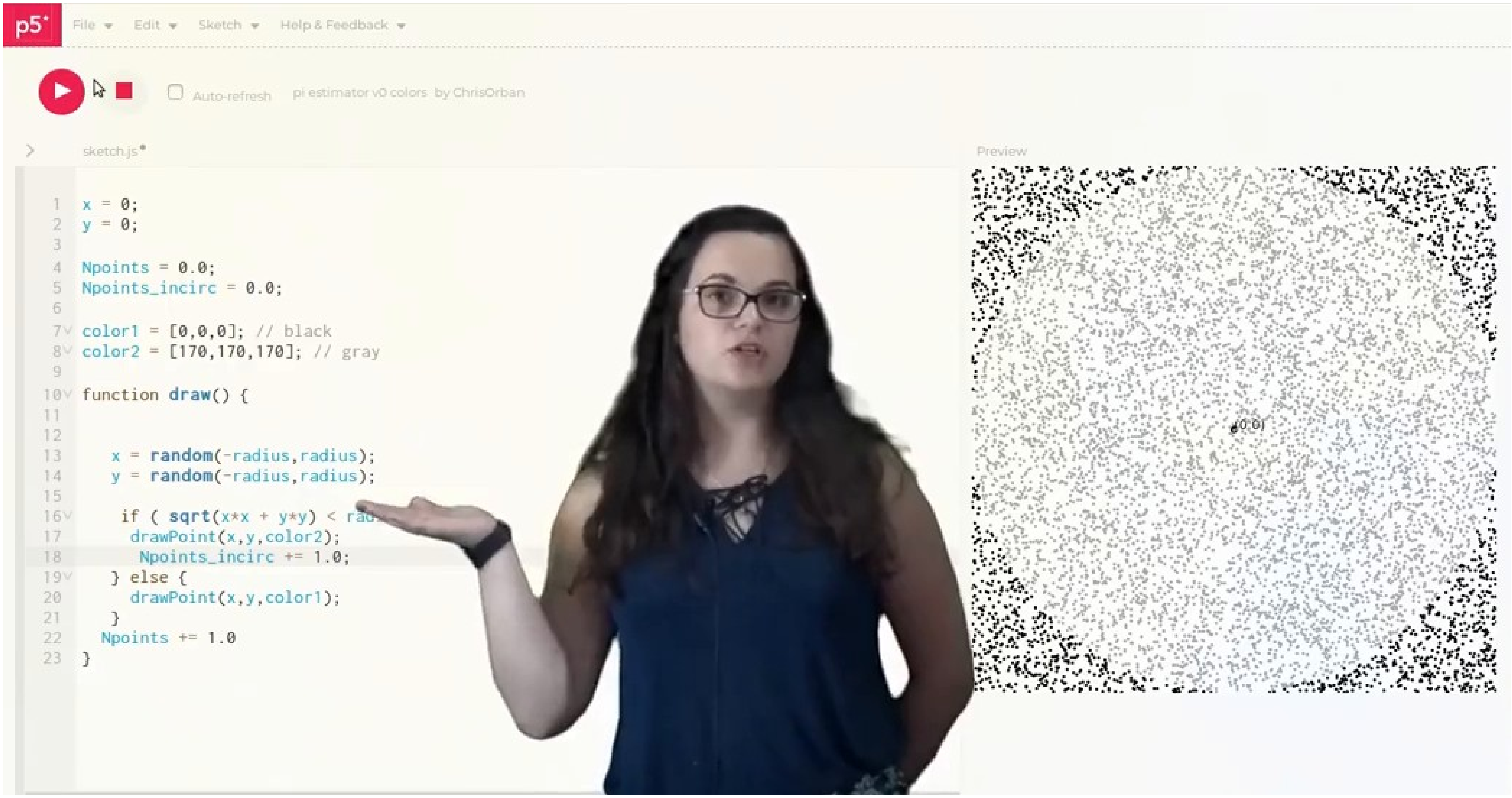
Step 1. Open the app. Press "+" to open a new e l:
in the lower right corner to edit the new experim bol:
at the bottom of the screen to bring up the vario

Step 2. Click on this symbol (fourth s At this
point your screen should look somethin

https://www.asc.ohio-state.edu/orban.14/stemcoding/pressure/

How would I connect with...

Math

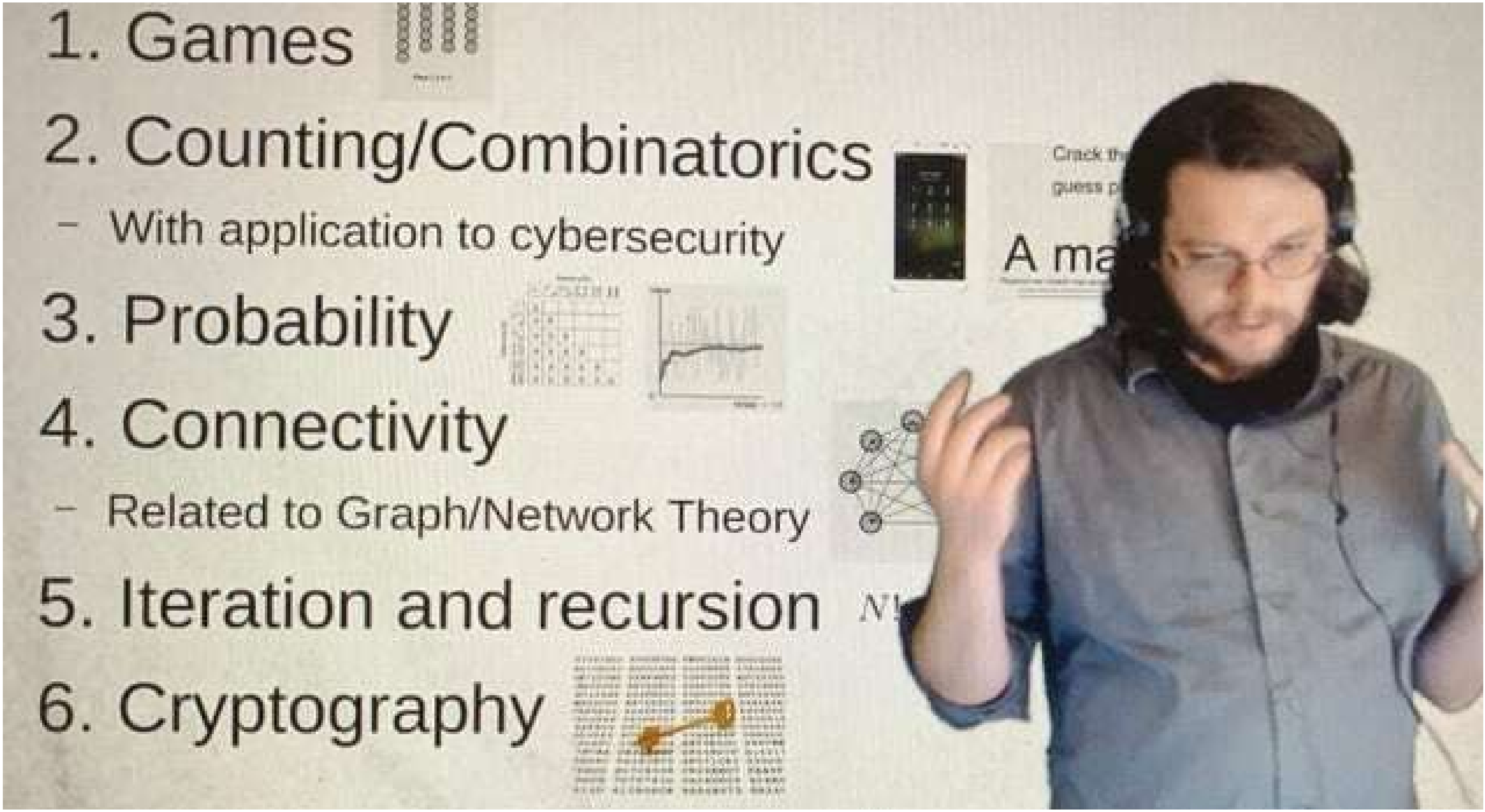


```
x = 0;
y = 0;

Npoints = 0.0;
Npoints_incirc = 0.0;

color1 = [0,0,0]; // black
color2 = [170,170,170]; // gray

function draw() {


   x = random(-radius,radius);
   y = random(-radius,radius);

    if ( sqrt(x*x + y*y) < rad
     drawPoint(x,y,color2);
      Npoints_incirc += 1.0;
   } else {
     drawPoint(x,y,color1);
   }
  Npoints += 1.0
}
```

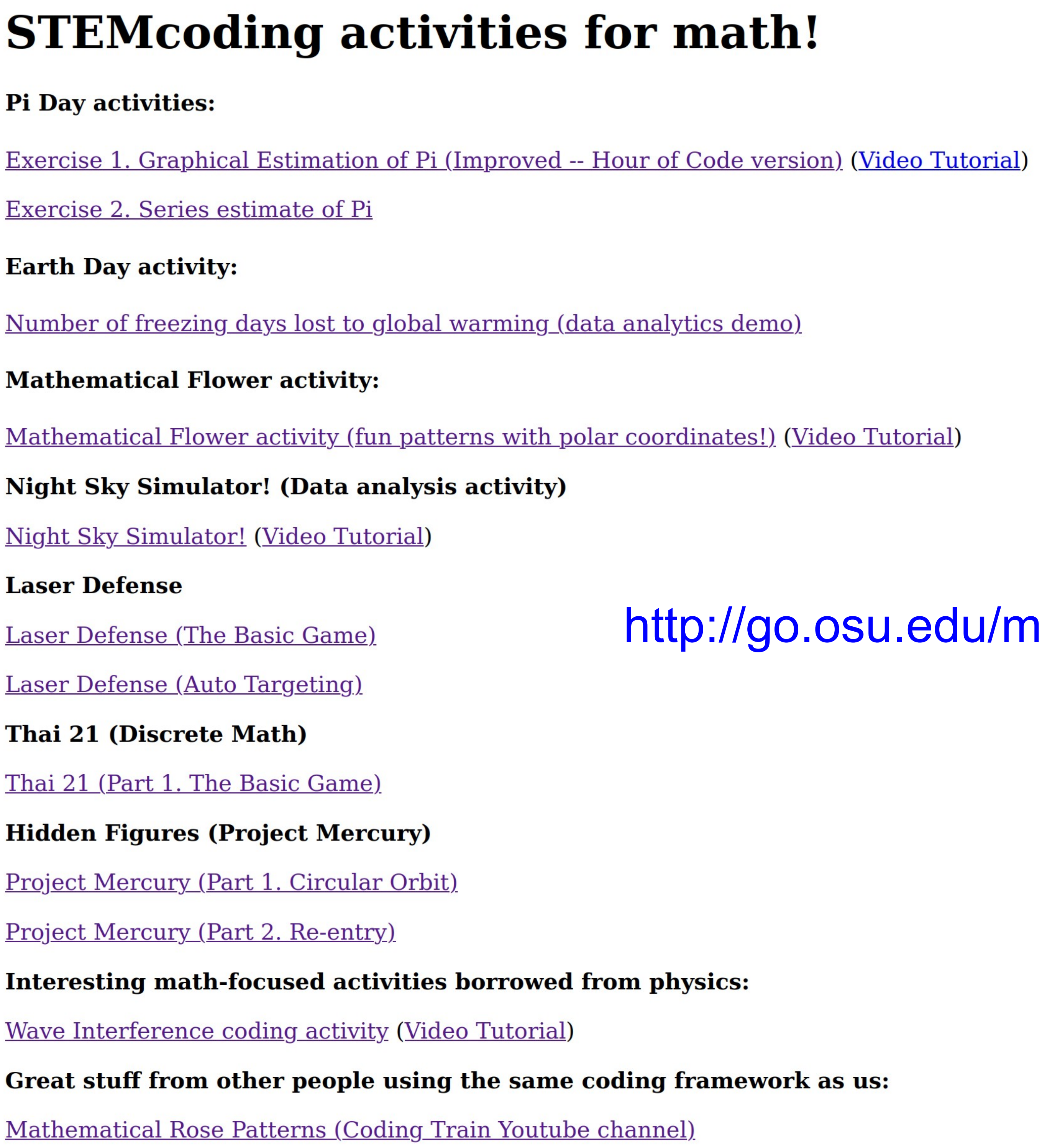
http://go.osu.edu/pidaycoding

STEM coding

Discrete math



1. Games
2. Counting/Combinatorics
   - With application to cybersecurity
3. Probability
4. Connectivity
   - Related to Graph/Network Theory
5. Iteration and recursion
6. Cryptography



STEMcoding activities for math!

Pi Day activities:

Exercise 1. Graphical Estimation of Pi (Improved -- Hour of Code version) (Video Tutorial)

Exercise 2. Series estimate of Pi

Earth Day activity:

Number of freezing days lost to global warming (data analytics demo)

Mathematical Flower activity:

Mathematical Flower activity (fun patterns with polar coordinates!) (Video Tutorial)

Night Sky Simulator! (Data analysis activity)

Night Sky Simulator! (Video Tutorial)

Laser Defense

Laser Defense (The Basic Game)

Laser Defense (Auto Targeting)

Thai 21 (Discrete Math)

Thai 21 (Part 1. The Basic Game)

Hidden Figures (Project Mercury)

Project Mercury (Part 1. Circular Orbit)

Project Mercury (Part 2. Re-entry)

Interesting math-focused activities borrowed from physics:

Wave Interference coding activity (Video Tutorial)

Great stuff from other people using the same coding framework as us:

Mathematical Rose Patterns (Coding Train Youtube channel)

http://go.osu.edu/math_coding

Questions?

orban@physics.osu.edu
stemcodingohio@gmail.com

go.osu.edu/objecttracker

youtube.com/STEMcoding

stemcoding.herokuapp.com

Monthly email list:
u.osu.edu/stemcoding

@STEMcoding

STEMcoding