Jasmine Truong

Berkeley Space Technologies and Rocketry Team (STAR)

Rainstorm Workshop Lesson Plan (25min)

5 minutes

1. Conceptual explanation of what an algorithm is
   1. Short PowerPoint Presentation
      1. Discuss how an algorithm is like a set of instructions (like a recipe)
      2. What happens when there is an error or bug in the code?
      3. How can you debug your code?
      4. Basic coding syntax and what they do: If, Else, While, Return, Event/Keyboard Listeners
   2. Questions?

5 minutes

1. Analogy for Algorithm (Warm-Up)
   1. Algorithm to tie shoes graphic organizer
      1. <https://www.brainpop.com/math/dataanalysis/computerprogramming/graphicorganizer/>
   2. Encourage the use of coding key words/syntax
      1. If, else, while, return
      2. Example:
         1. If shoes are not already tied: take both laces
         2. Else: exit, no need to tie shoes
         3. While both laces are in hand: create two bunny ears and twist together, loop into knot, and tighten
         4. Return: Shoes are tied!

15 minutes

1. Create Algorithm to Help Character Escape the Maze in Codesters Using Custom Project:
   1. Moving character manually will help you solve a specific maze
      1. Running algorithm at every step can be tedious, but it can also prevent errors and bugs you have to fix later!
   2. More Advanced: How can you solve for a generic maze with user input?
      1. Using event/keyboard listeners where the person playing can move the sprite themselves
2. Codesters Solution Key

1. #YOU CHANGE MAZE BY SEARCHING 'MAZE' IN STAGE

stage.set\_background("maze2")

def start\_sprite(sprite):

sprite.set\_left(-280)

sprite.set\_top(300)

size = sprite.get\_size()

if size >= 1:

sprite.set\_size(0.5)

sprite.pen\_down()

sprite.pen\_width(2)

##CHANGE PEN COLOR HERE (Any color of the rainbow)##

sprite.set\_color("green")

#REPLACE LINE 16 WITH YOUR SPRITE##

sprite = codesters.Sprite("alien1")

start\_sprite(sprite)

##START YOUR ALGORITHM!##

#MANUAL SPRITE ACTION (WORKS FOR A SPECIFIC MAZE)#

sprite.move\_right(140)

sprite.move\_down(50)

sprite.move\_right(100)

sprite.move\_up(50)

sprite.move\_right(100)

sprite.move\_down(110)

sprite.move\_right(100)

sprite.move\_down(40)

sprite.move\_left(150)

sprite.move\_down(50)

sprite.move\_right(150)

#EVENT LISTENER (WORKS FOR ANY MAZE)#

def left\_key():

sprite.move\_left(20)

# add other actions...

stage.event\_key("left", left\_key)

def right\_key():

sprite.move\_right(20)

# add other actions...

stage.event\_key("right", right\_key)

def down\_key():

sprite.move\_down(20)

# add other actions...

stage.event\_key("down", down\_key)

def down\_key():

sprite.move\_down(20)

# add other actions...

stage.event\_key("down", down\_key)

def up\_key():

sprite.move\_up(20)

# add other actions...

stage.event\_key("up", up\_key)