OAK RIDGE COMPUTER SCIENCE GIRLS

ORCS Girls

BLOUNT COUNTY PUBLIC LIBRARY

UT-BATTELLE

SMART ROBOTICS CAMP

ORCSGirls

ORCSGirls

RULES

- Treat each other, the volunteers and the facilities with respect.
- Raise your hand if you have a question or want to answer.
- Only visit the websites needed for the class project.
- Always follow instructions by volunteers.
- Wait in the room until you are picked up at the end of the day.
- **Be gentle with the robotic arms, connectors, and monitors!**
- Help each other.
- Enjoy!

Introductions

Welcome to our camp

Instructors and Volunteers

Thomas Proffen

Amelie Nagle

Grace Feng

Isabel Kao

Rachel Burket

Yvonne Dalschen

GETTING TO KNOW EACH OTHER BINGO

ORCS Girls BINGO

Visited another country	Never had a cavity	Knows how to knit or crochet	Ridden a horse	Loves fast-food
Eaten an insect	Never broken a bone	Skipped breakfast today	Watches reality TV	Likes anchovies on pizza
Rides a bike to school	Went to a summer camp	 Free	Swam in a lake	Has more than three siblings
Slept in an airport	Flown in a helicopter	Eaten duck	Owns a cat	Does not know how to swim
Been to Hawaii	Cleaned a bathroom this week	Change a diaper	Plays a musical instrument	Gets up before 6 a.m.

Smart Robotics Camp

Schedule

Monday	Tuesday	Wednesday	Thursday
Welcome Driver / Navigator Setting up JetMax Control via webpage	More coordinates and inverse kinematics Python introduction Linux introduction	AI introduction ROS Operating System Calibration	Python external display module AI color demos
Lunch	**Lunch**	**Lunch**	**Lunch**
Coordinates Measure and move activity Block stacking action recording	Moving JetMax using Python Measuring and block stacking using Python	Block stacking demo Recycling classification and sorting	Checkout the code Modify code to show angle on external display 😯

Smart Robotics Camp

This is an *Advanced* Camp

Things to remember

- You got this!
- Do not give up if things seems hard in the beginning.
- Ask questions.
- All of us ‘google’ things all the time.
- Nobody writes code that works right away.
- Debugging can be frustrating, but it is super sweet if things work in the end.
- We have a **flexible schedule**.
- Have fun!

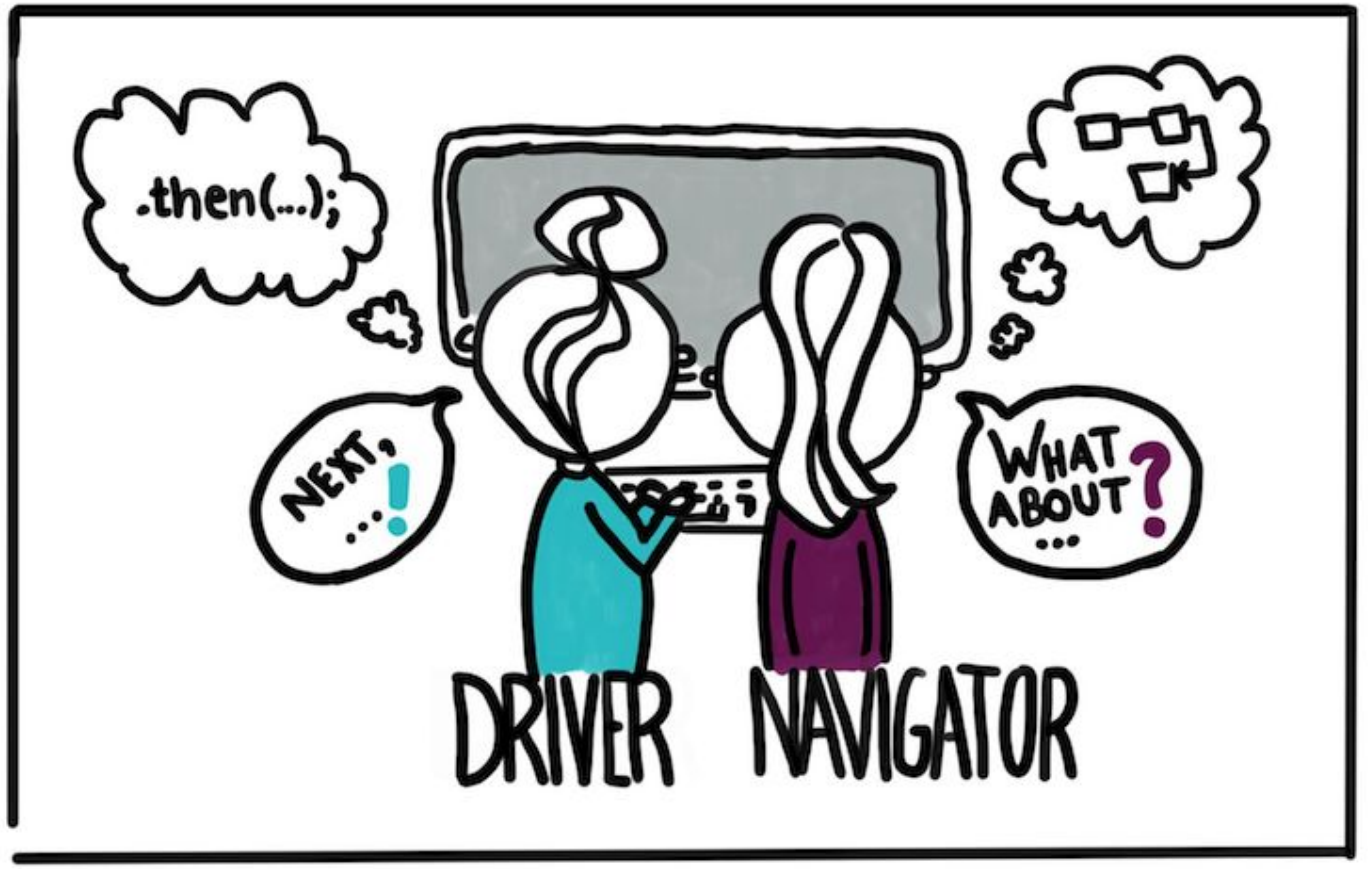
Pair Programming

Two girls

One laptop

One robotic arm

Unstoppable



Driver

- **Write the code or run commands according to the navigator's specification**
- **Do not keep going just because you know what is next.**
- Listen intently to the navigators instructions
- Ask questions wherever there is a lack of clarity
- **Offer alternative solutions if you disagree with the navigator**
- If there is disagreement, defer to navigator. If idea fails, get to failure quickly and move on
- **Own the computer / keyboard**
- **Trust the navigator -** ultimately the navigator has the final say in what is done
- You are writing the code, clicking the buttons and run commands

Navigator

- **Dictates the code that is to be written or commands to be run - the 'what'**
- Clearly communicates what code to write or what to enter
- **Explains 'why' they have chosen the particular solution to this problem**
- Check for syntax / type errors as the Driver drives
- **Be the safety net for the driver**
- Make sure that the driver sticks to the small task at hand
- **Ongoing code review**
- Pay attention

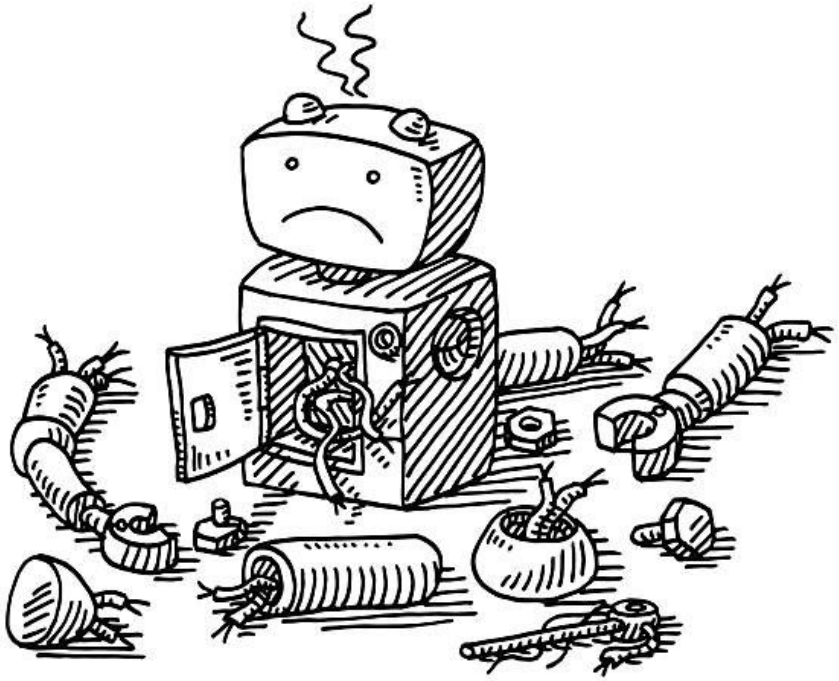
Both

Working together is the key!

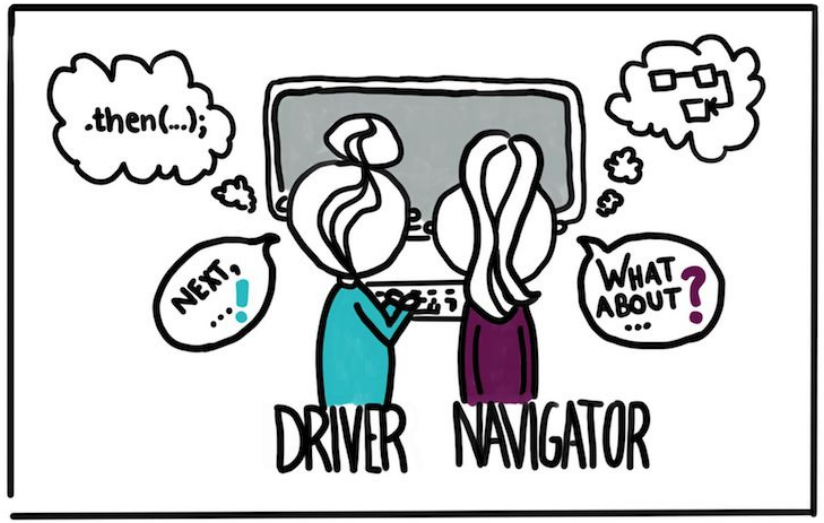
- **Actively take part in programming and operating the arm**
- Aim for optimal flow - avoid trying to be 'right'
- Embrace your role
- Intervene if your pair is quiet
- Know when to swap roles or take a break
- **Communicate, communicate, COMMUNICATE!**
- **Don't hog the keyboard**
- **High-five every time something works high**
- **Swap roles frequently**

Robot rules

Take turns - communicate - stay engaged - don't fight over who does what!



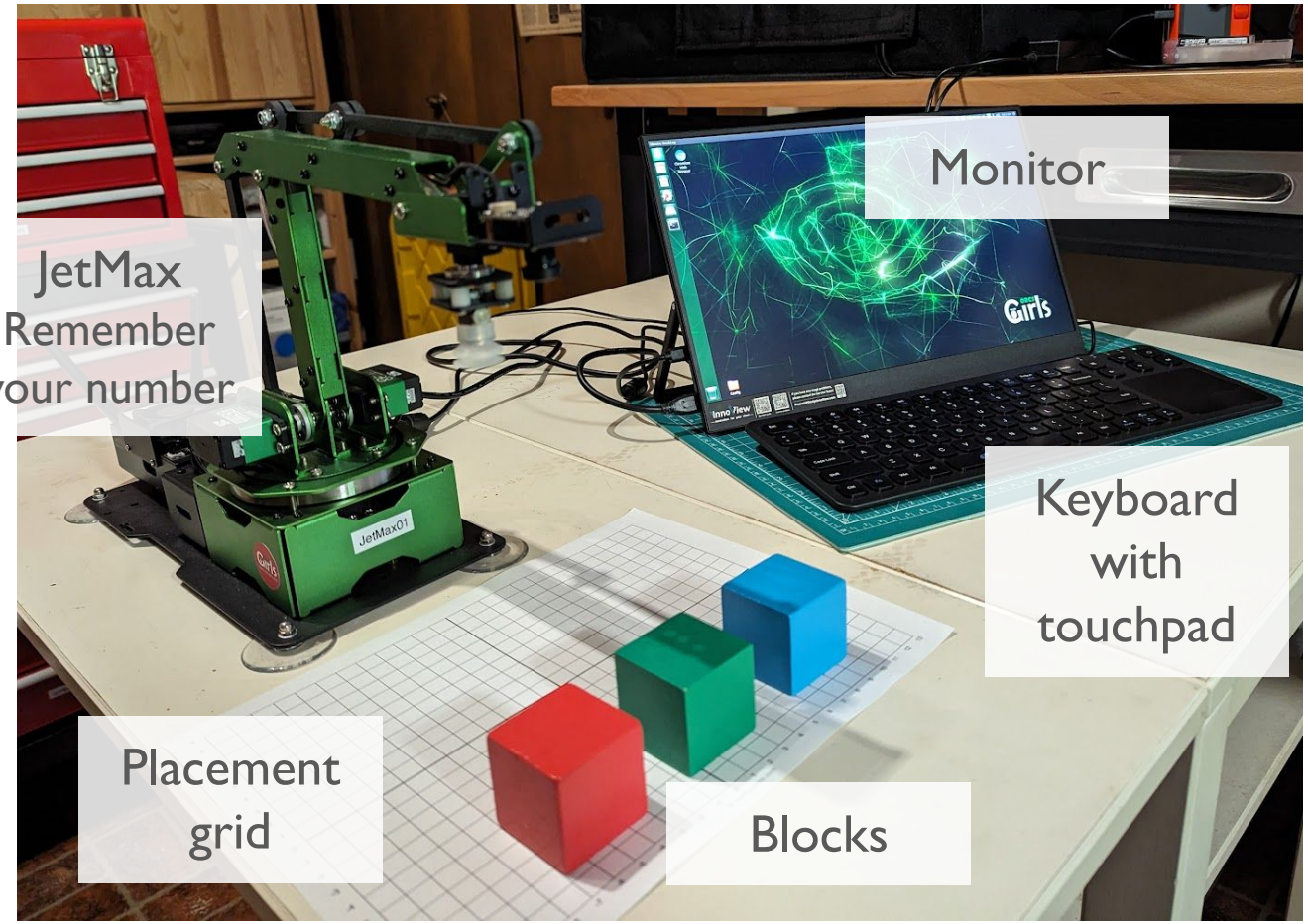
- Only pickup robotic arm by its base
- Be gentle when plugging in cables
- Do not move the arm manually
- Use proper shutdown before powering off
- Use slow speed for moving
- Be patient, commands might take a while
- Double check your coordinates are right before moving, so the arm does not collide with something
- Keep your work area clean



Let's pair up

We keep these for the entire camp.

Setting up the JetMax



Clear a workspace

Decide who picks up what

Volunteers will hand our materials

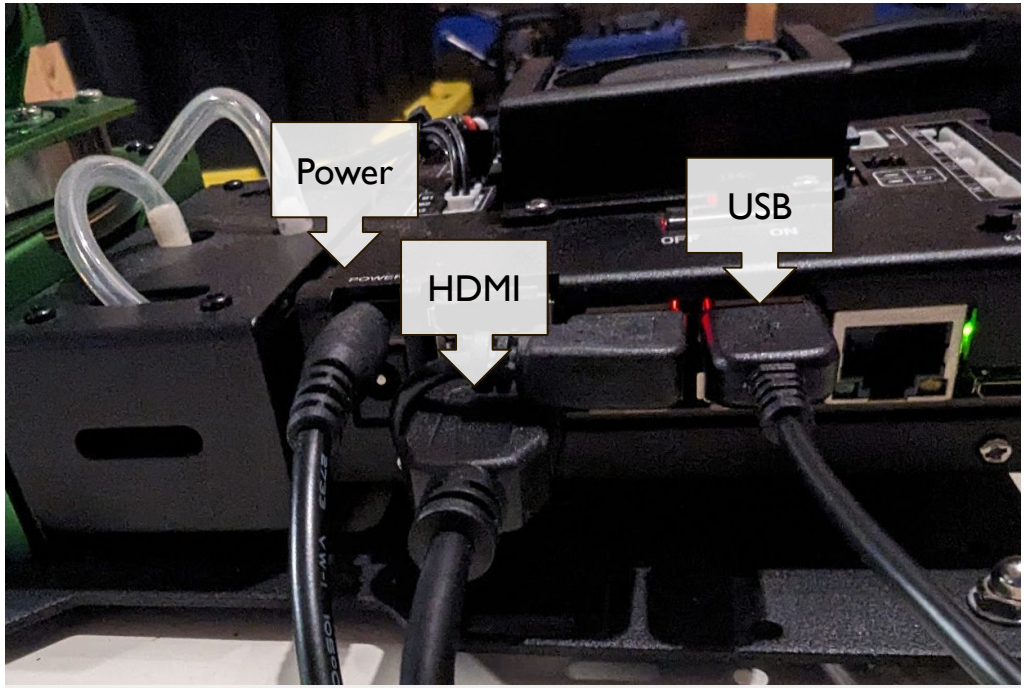
You need

- JetMax robot arm
- Monitor
- Keyboard
- Placement grid
- Three color blocks
- Power supply JetMax
- Power supply monitor
- HDMI cable

Smart Robotics Camp

Connecting stuff ..

Be **gentle** with the connectors



Make sure JetMax switch is turned off

Connect power (careful, this might be hard to get in, ask for help if needed)

Connect HDMI cable from monitor

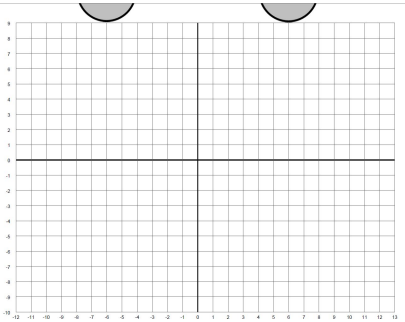
Connect USB connector from keyboard

Connect USB-C from power supply

Connect HDMI cable going to JetMax

Place grid in front of JetMax

Match front suction feet with marks on sheet.

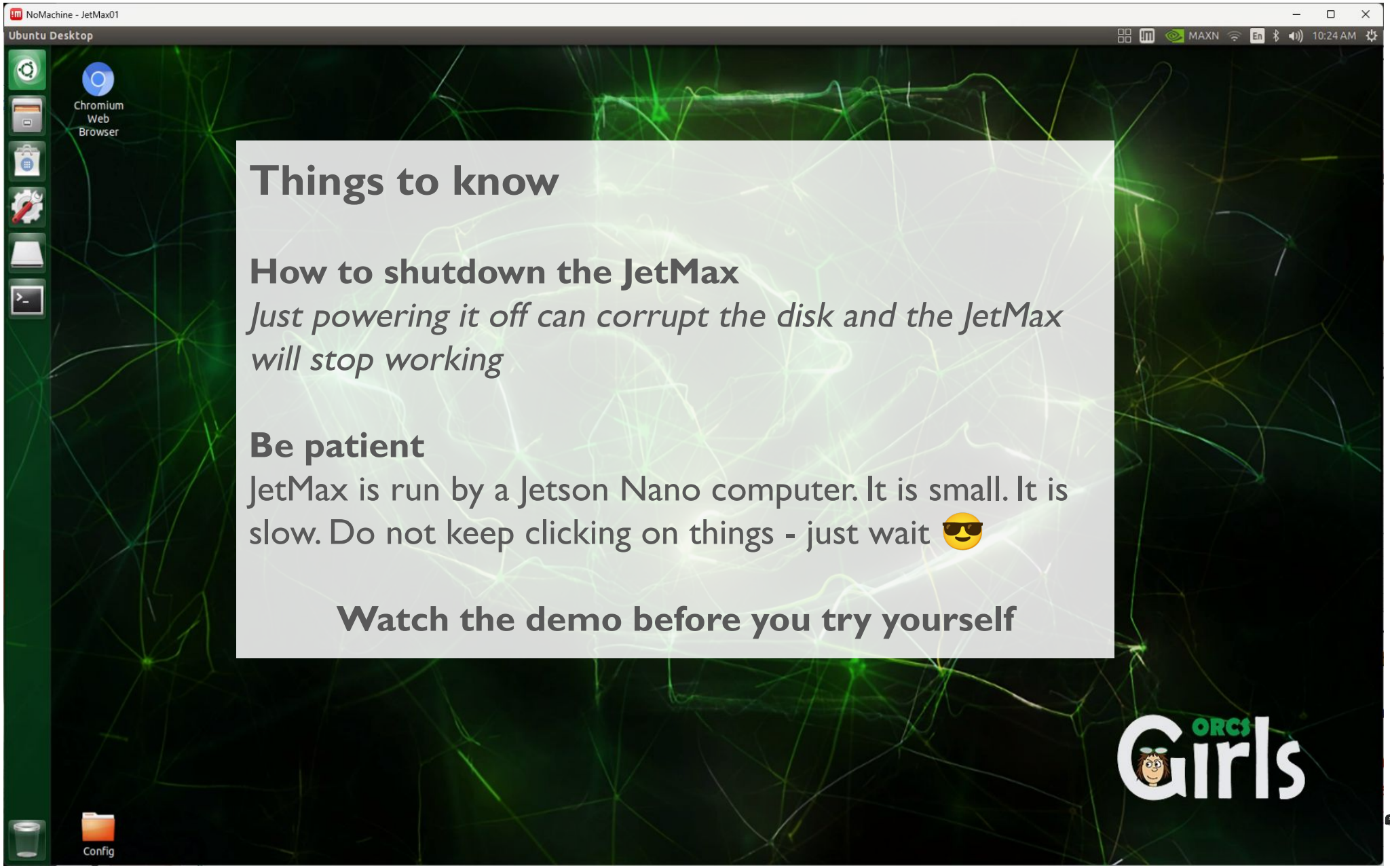


Turn the JetMax and monitor on

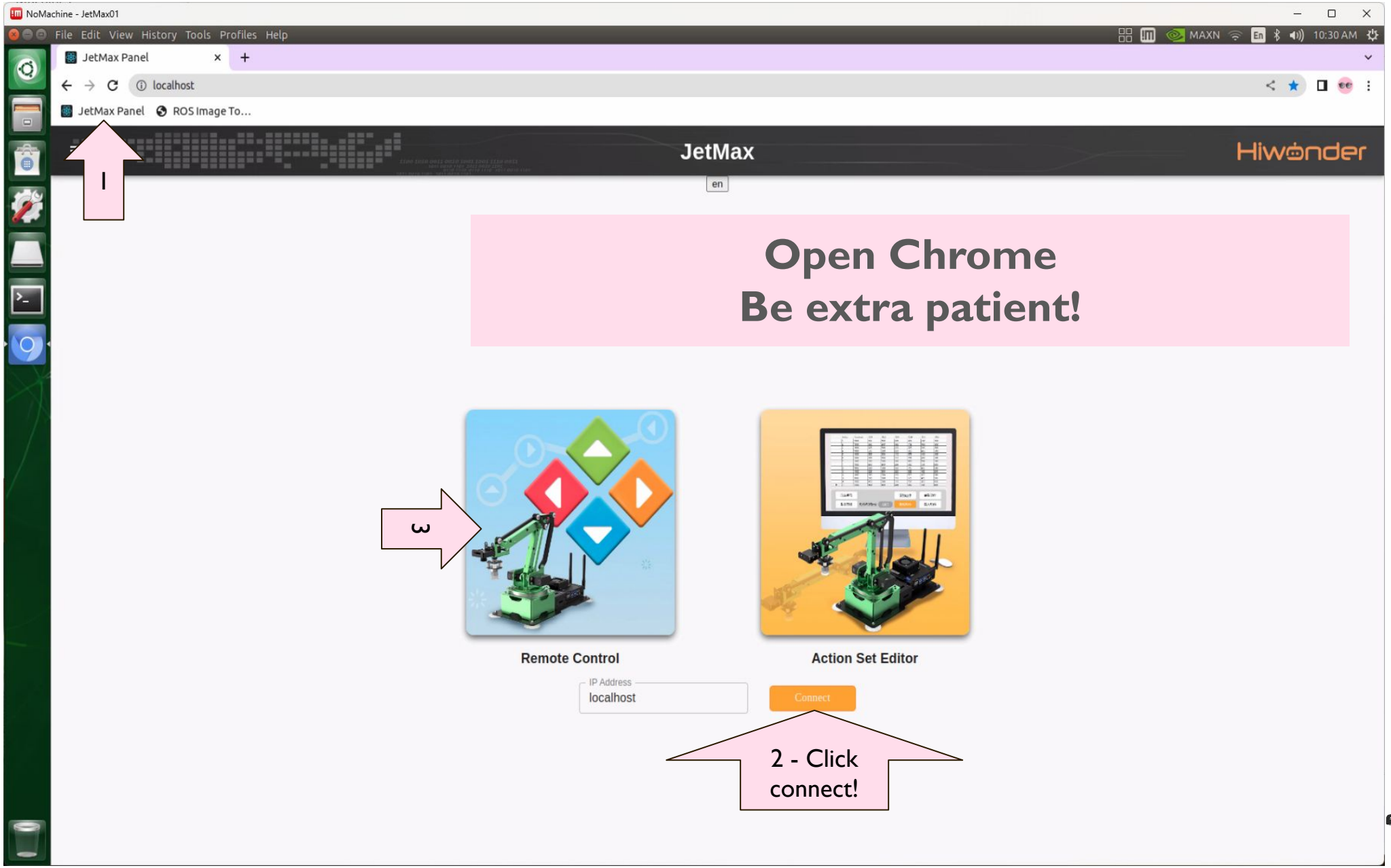
Wait for the beep

WAIT
We will start together

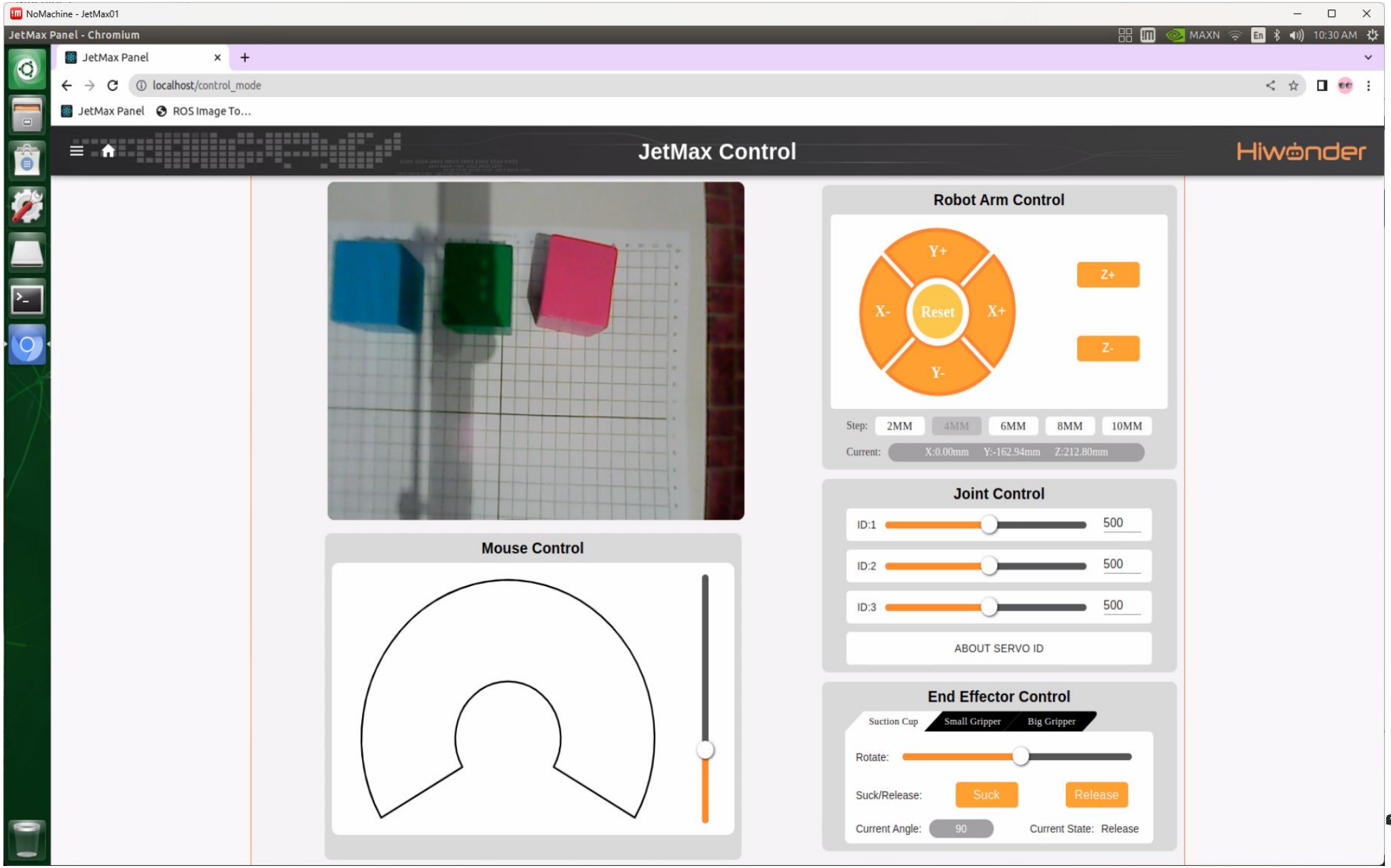
Smart Robotics Camp



Smart Robotics Camp

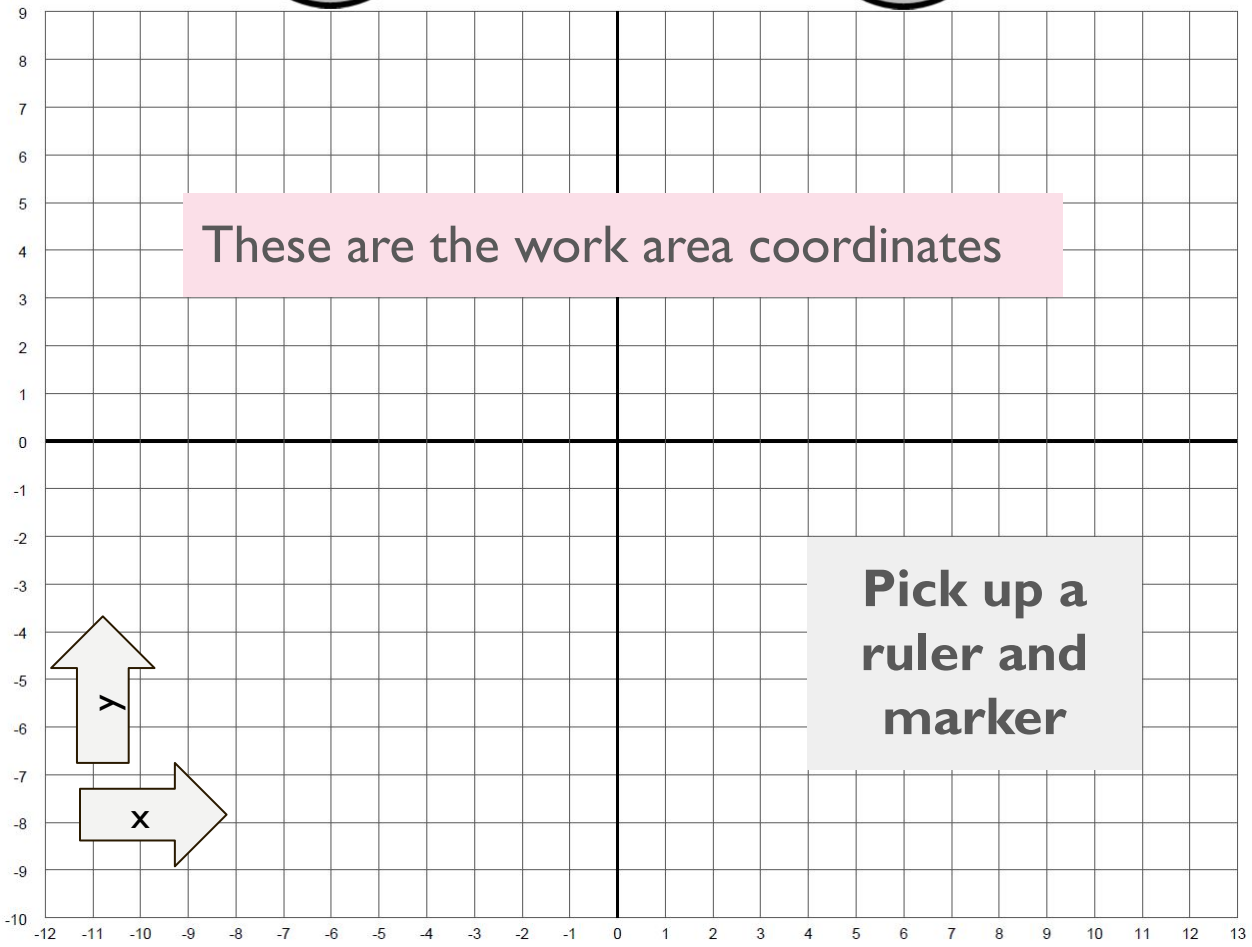


Smart Robotics Camp



Smart Robotics Camp

Smart Robotics Camp



These are the robot coordinates

Measure and Move Activity

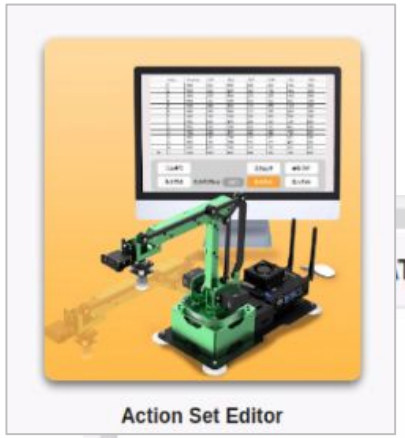
How to convert from robot to work area coordinates?

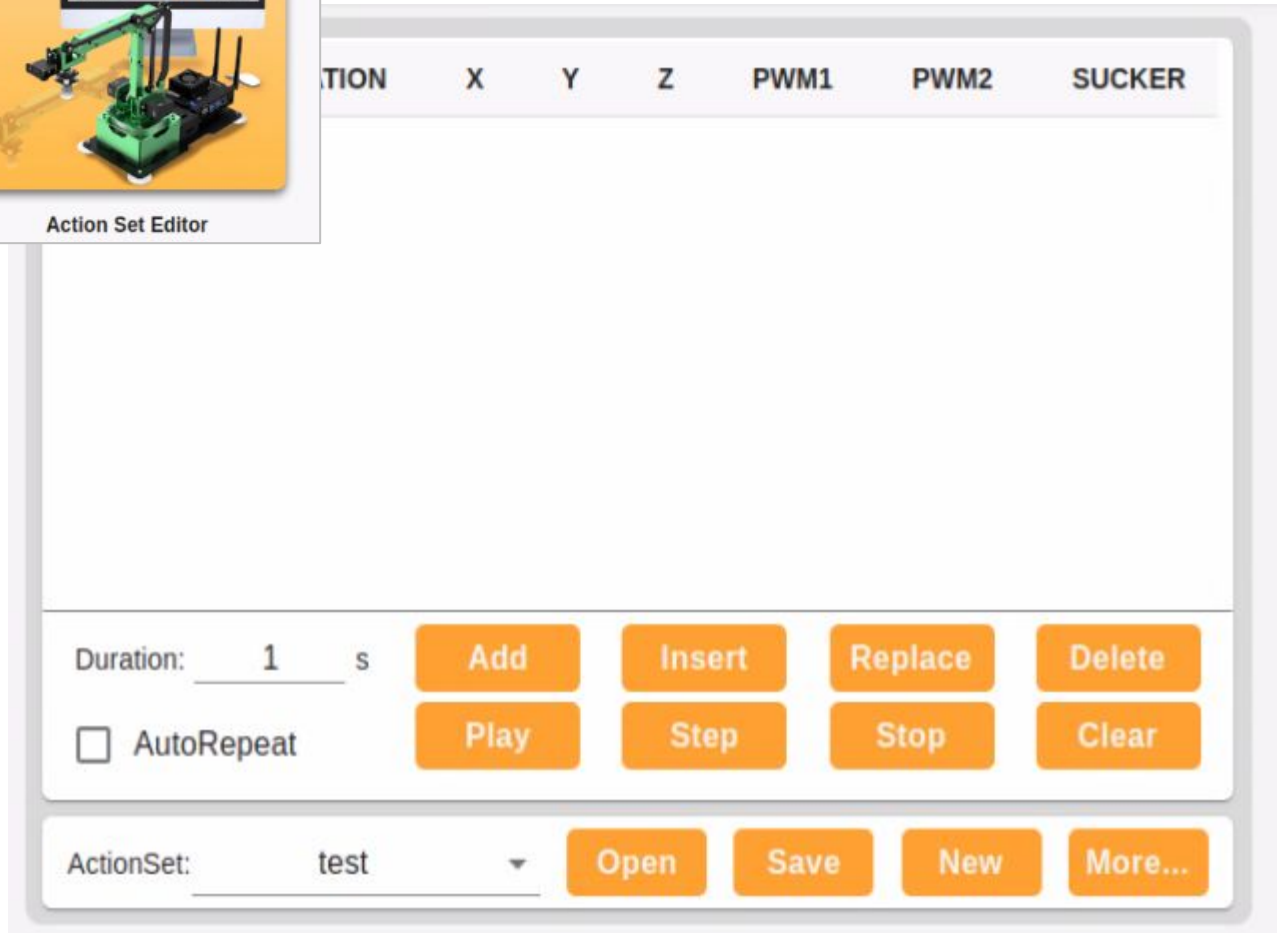
Where is (0,0,0) on the robot arm?

Challenge: put a block on the grid and determine robot coordinates

Check by driving there

Smart Robotics Camp





Do not forget to **save** and have a different name for each of you.

Stacking Challenge

Start the Action Set Editor

You can now save positions of the arm and replay them.

Play around

Challenge

- ➔ Put three blocks on the grid and mark the position
- ➔ Learn the actions to stack them.
- ➔ Put blocks back and replay - does it stack the blocks?

Showtime

Let’s check out who stacks the best 

That's it for today

- **Shut down the JetMax arm properly**
- Wait for the screen to go blank
- Switch the arm off
- Carefully disconnect cables
- Being all parts up and hand them to the volunteers
- **Remember your number, so you have the same robotic arm next time**
- See you tomorrow 🎉

OAK RIDGE COMPUTER SCIENCE GIRLS

ORCS Girls

BLOUNT COUNTY PUBLIC LIBRARY

UT-BATTELLE

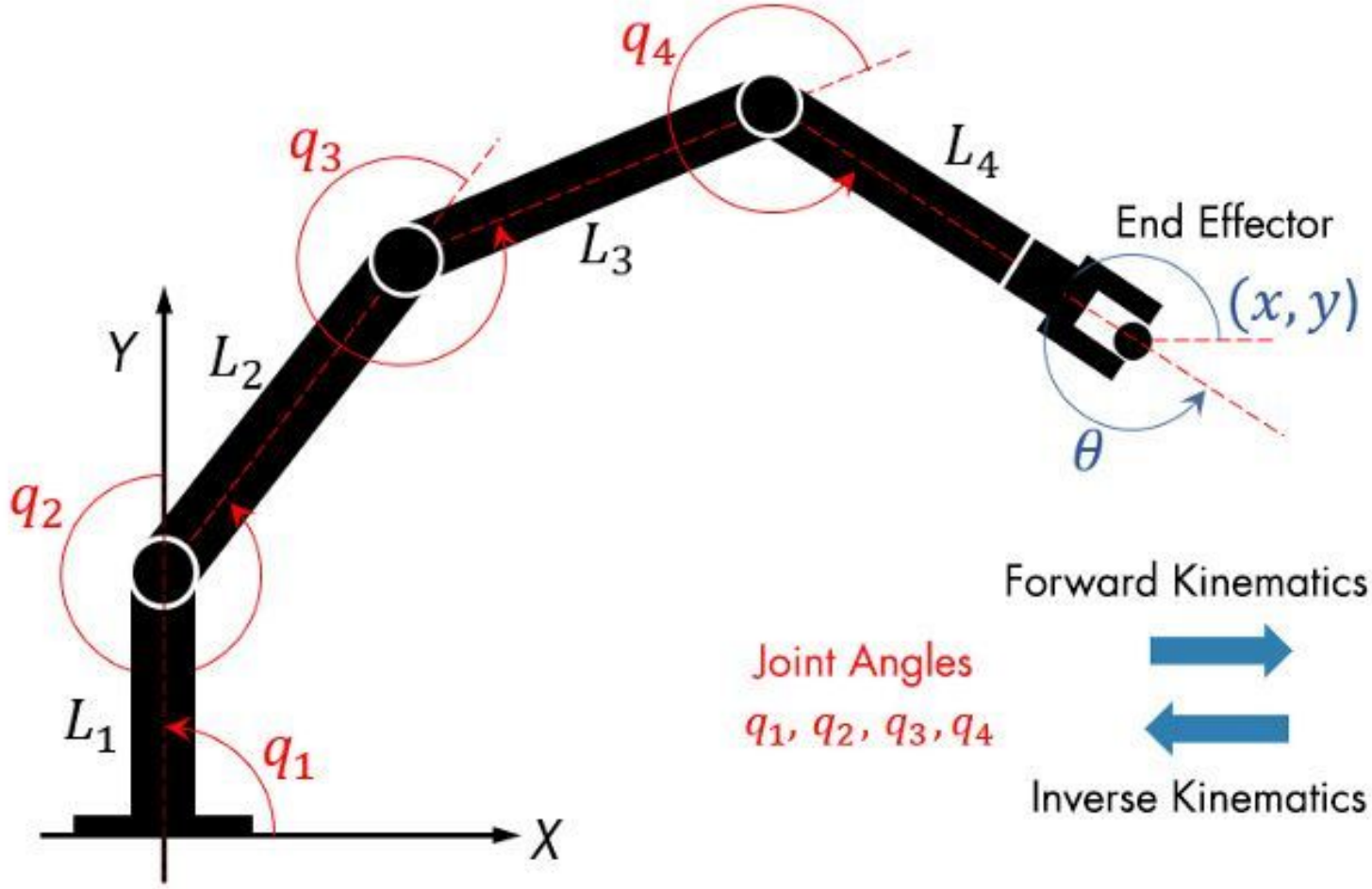
SMART ROBOTICS CAMP
Day 2

ORCSGirls

ORCSGirls

Review what we learned

Kinematics



Forward Kinematics (FK)

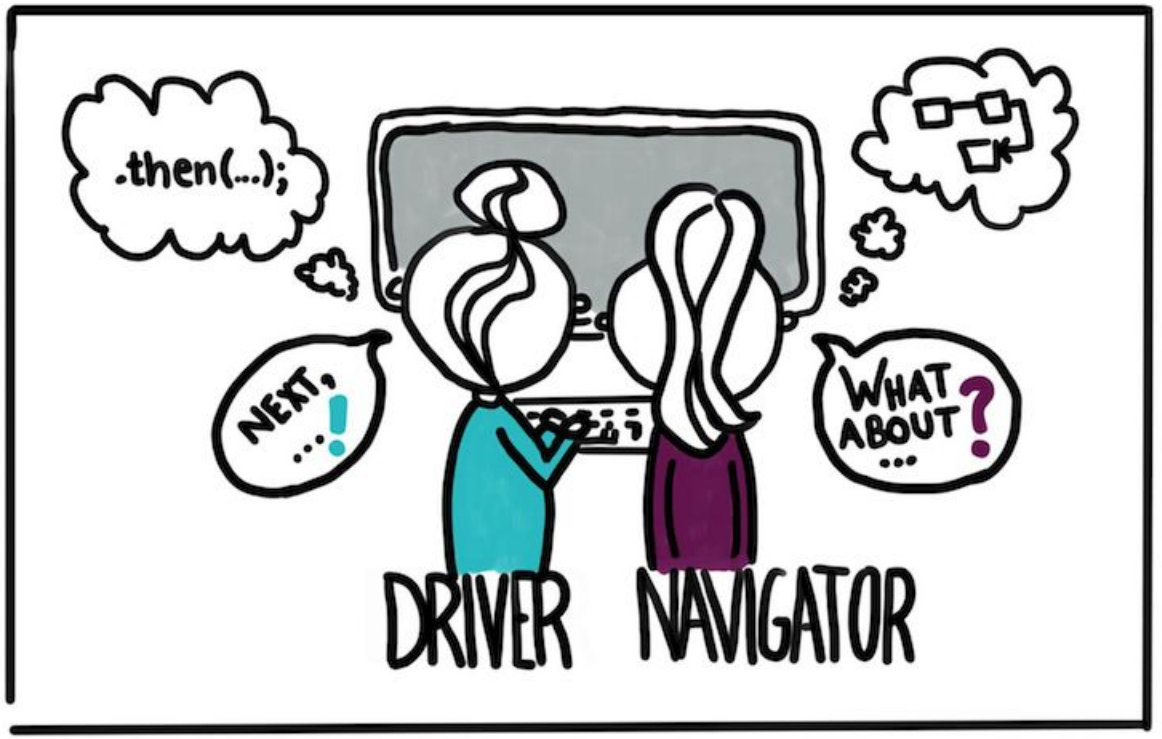
Joint Angles
q_1, q_2, q_3, q_4

End Effector Pose
x, y, θ

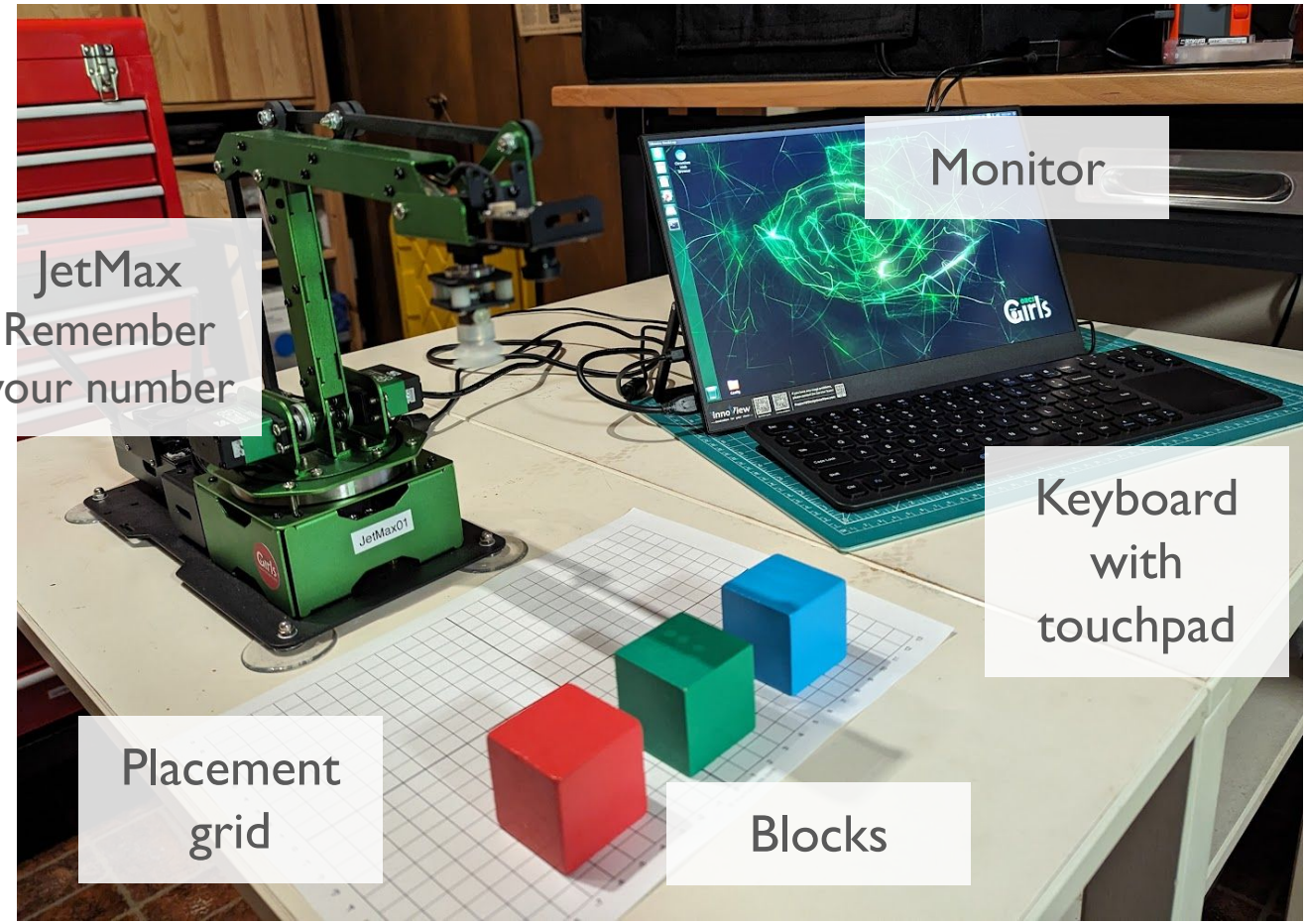
Inverse Kinematics (IK)

Remember *Driver Navigator* rules

- Make sure everyone gets to perform every task
- Take turns in setting up
- Help each other
- Listen and communicate
- Great teamwork leads to great progress and results
- Valuable skill to learn



Setting up the JetMax



Clear your workspace

Decide who picks up what

Volunteers will hand our materials

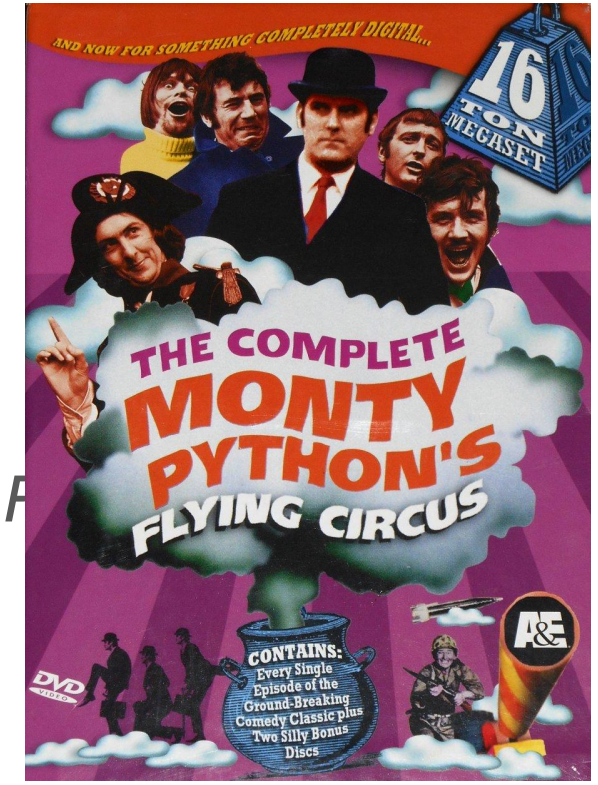
You need

- JetMax robot arm
- **Use same robot number**
- Monitor
- Keyboard
- Placement grid
- Three color blocks
- Power supply JetMax
- Power supply monitor
- HDMI cable

Smart Robotics Camp

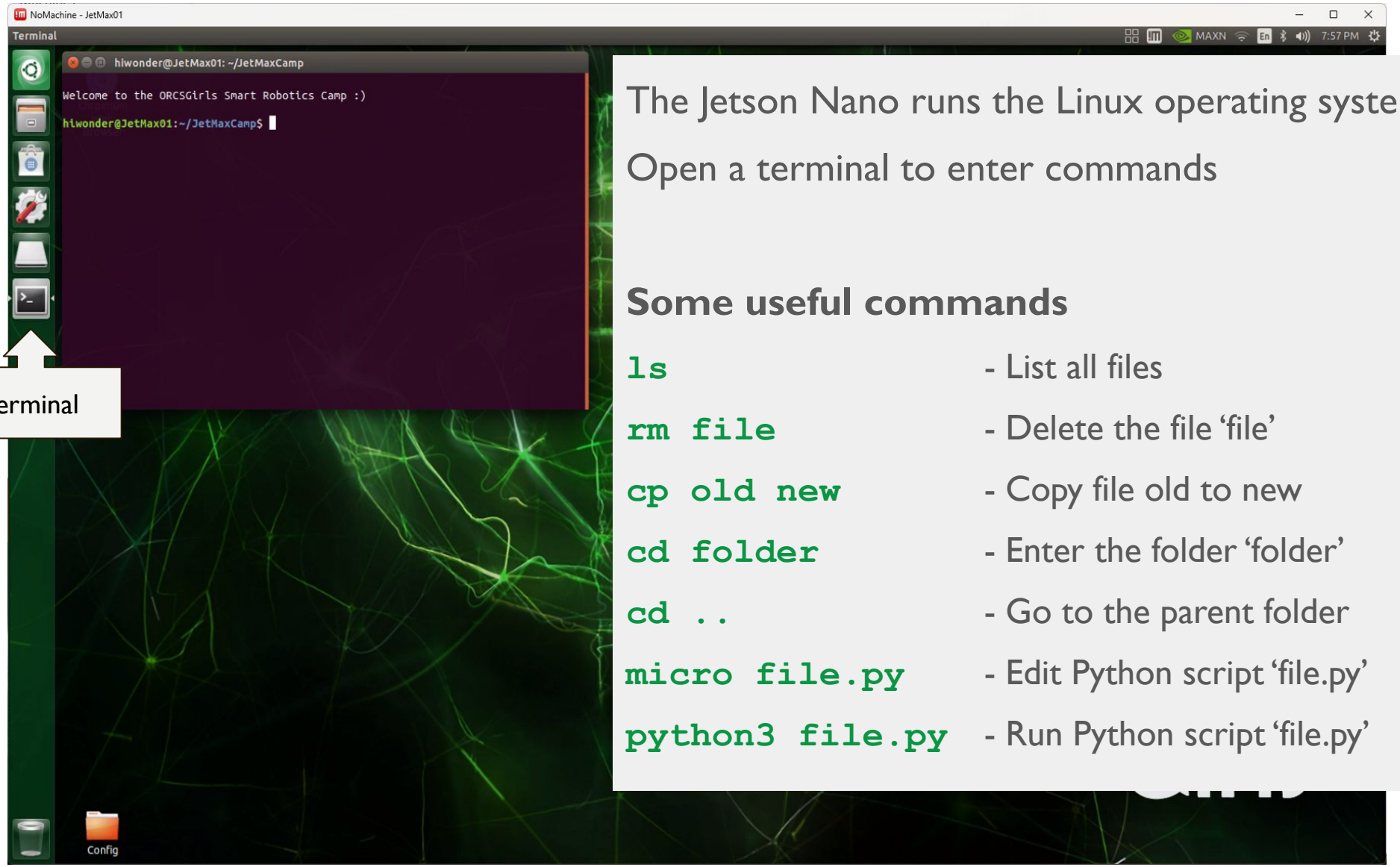
WHAT IS PYTHON?

- Python is an interpretive programming language.
- Python was created by Guido van Rossum in 1989.
- Named after his love for the comedy series *Monty P* *Circus* – not the snake.
- Benefits
  - Easy to learn.
  - Free to use and open source.
  - Portable, extensible, and embeddable.
  - Object oriented language.
- Fastest growing programming language.



Source: Wikipedia

Using the terminal



The Jetson Nano runs the Linux operating system

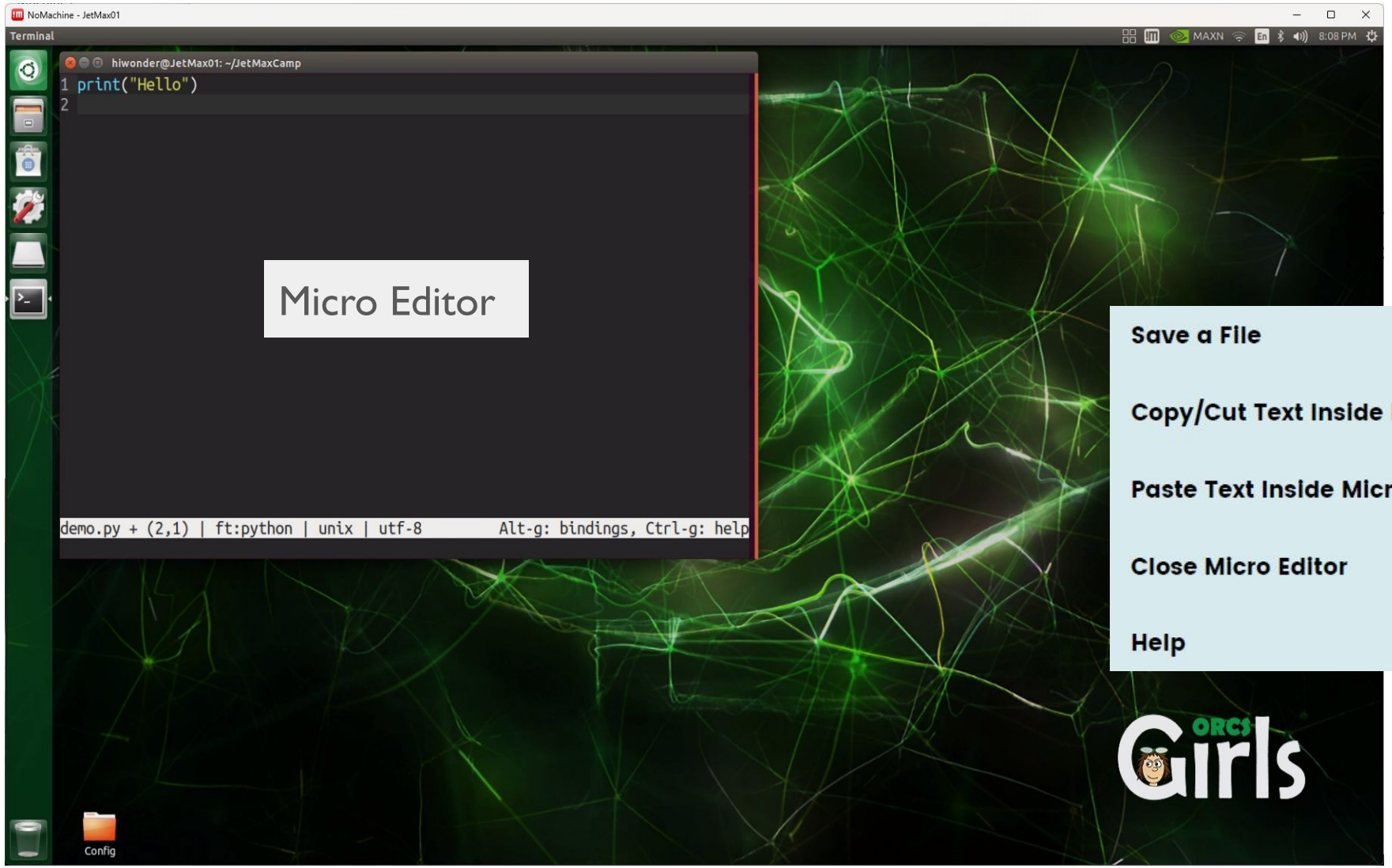
Open a terminal to enter commands

Some useful commands

`ls`	- List all files
`rm file`	- Delete the file 'file'
`cp old new`	- Copy file old to new
`cd folder`	- Enter the folder 'folder'
`cd ..`	- Go to the parent folder
`micro file.py`	- Edit Python script 'file.py'
`python3 file.py`	- Run Python script 'file.py'

Smart Robotics Camp

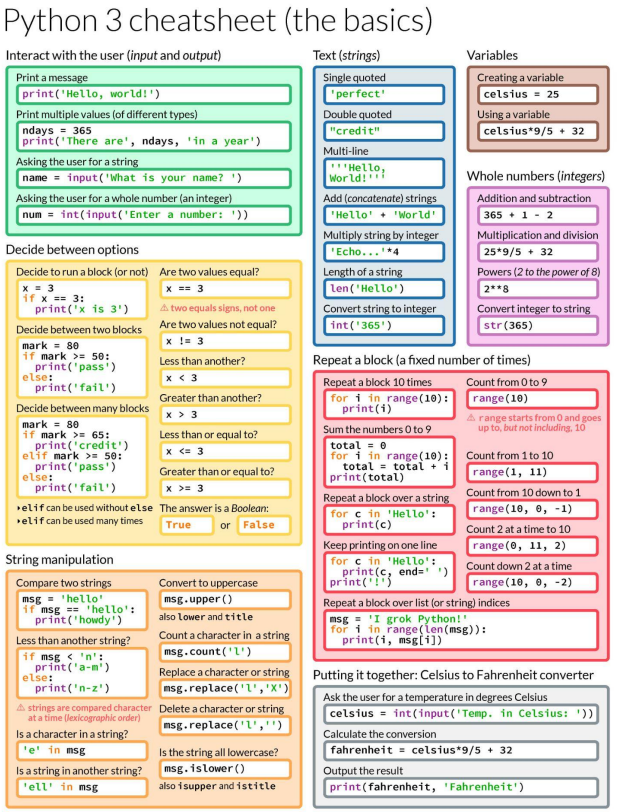
Using the terminal



Write our first code

Remember to takes turns who drives and who navigates

- Open a terminal
- Change to the Python folder - **`cd Python`**
- Create a new file and open in editor - **`micro first.py`**
- In the editor type - **`print("This camp is awesome")`**
- Save (control s) and quit the editor (control q)
- Run your code with **`python3 first.py`**
- Edit the code and make changes - explore Python.
- Brainstorm some simple programs together, e.g. print numbers from 1 to 10 or try the temperature converter from the handout)



Use the handout

We will start together

Smart Robotics Camp

Controlling the robot in Python

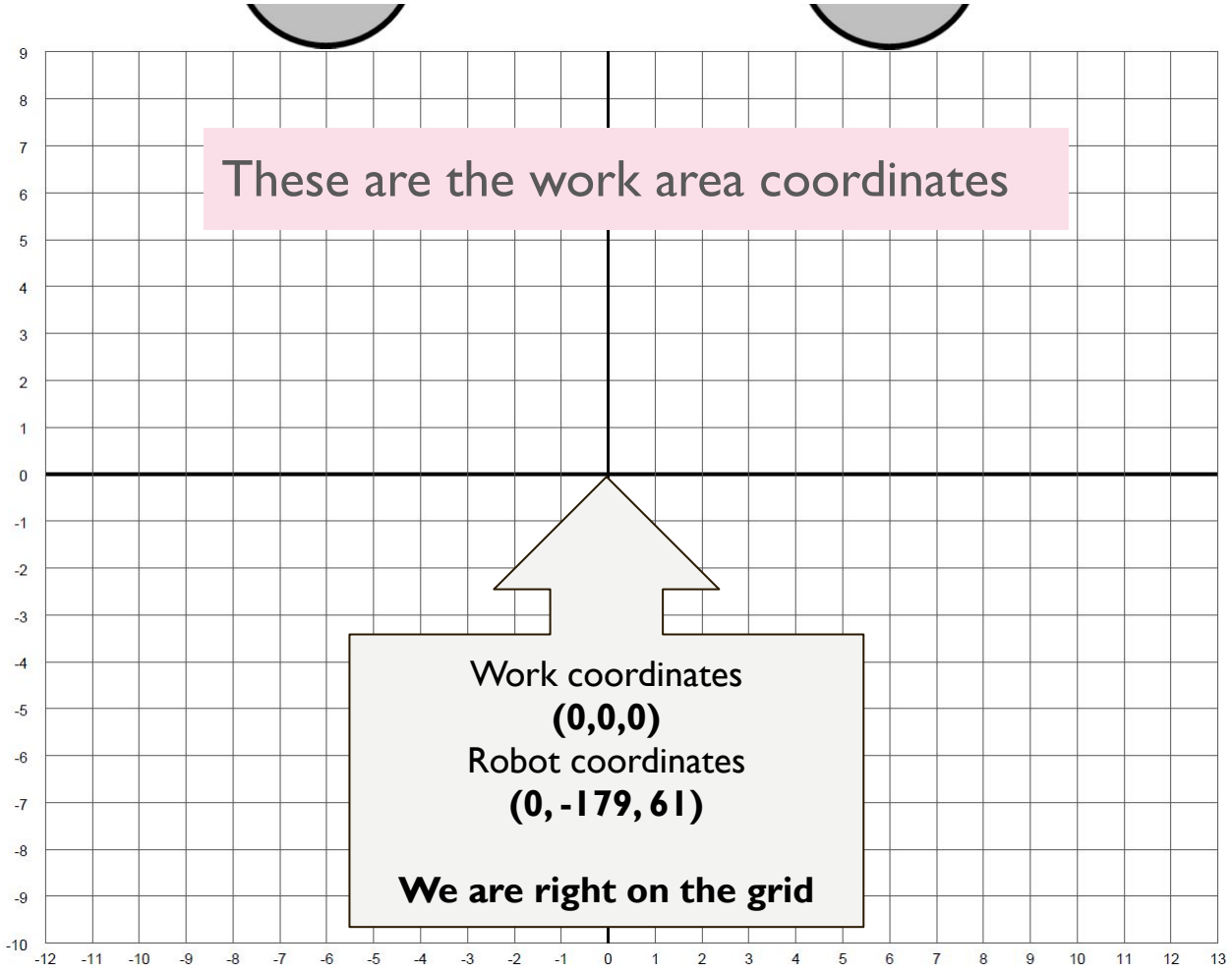
```
jetmax = hiwonder.JetMax()
sucker = hiwonder.Sucker()

jetmax.go_home()
cur_x, cur_y, cur_z = jetmax.position

jetmax.set_position((x,y,x), time)
jetmax.set_position_relatively((dx,dy,dz), time)

sucker.suck()
sucker.release(time)
```

JetMax Commands

What is the difference between `set_position` and `set_position_relatively`?

- Open a terminal
- Go to the Python folder
- Run the **move.py** program
- Edit the program and try different moves
- **Can you pick up a block?**

Drive to work area coordinates



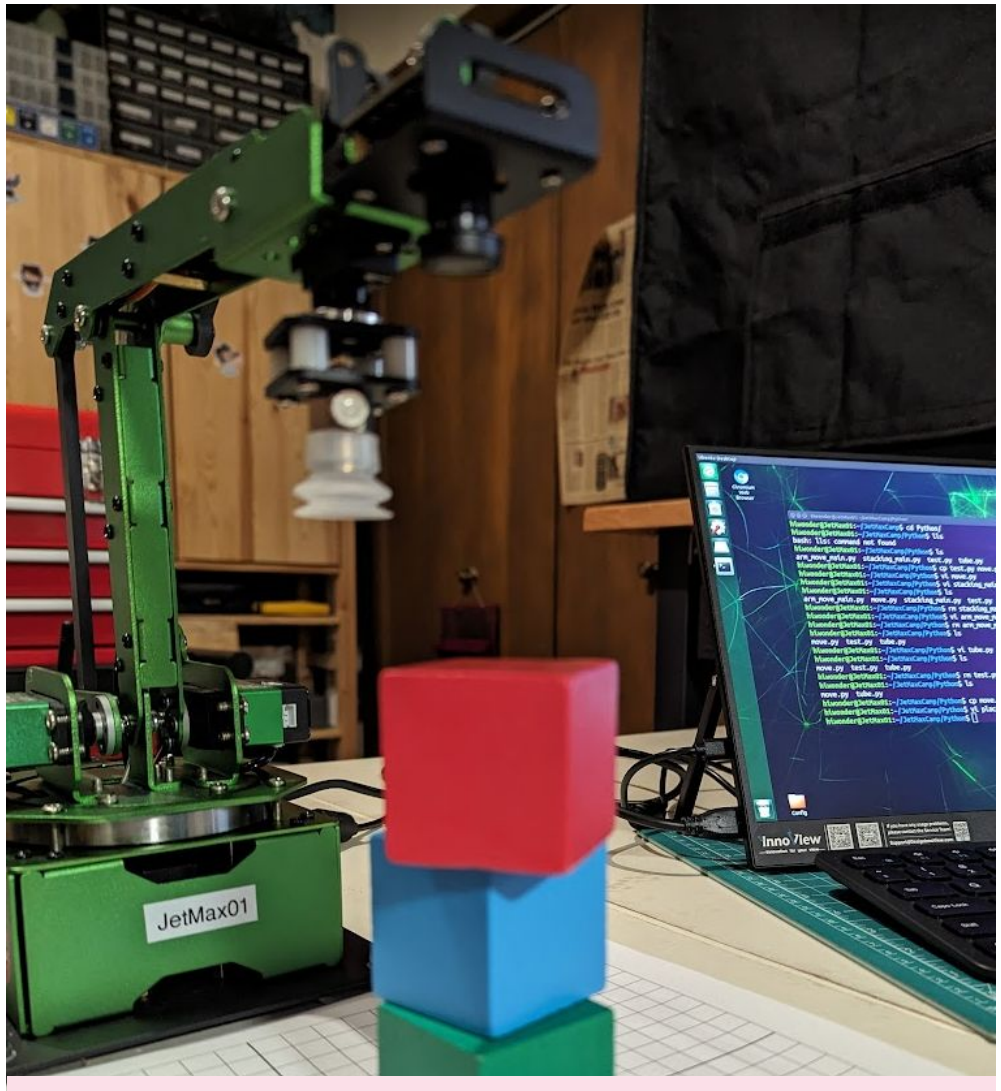
How can we convert from work area coordinates to robot coordinates?

We want to write code so we can tell the arm to goto a point on the grid - for example (0,0) in the middle

Let's brainstorm first

Smart Robotics Camp

Smart Robotics Camp



Make sure the grid mat does not move.

Remember to use Driver Navigator rules when writing and testing the code!

JetMax Coding Challenge

- ➔ Select two locations on the grid and note there x and y
- ➔ Measure the height of a block
- ➔ Put a block on the start location
- ➔ You program should pickup the block and move it to the other location

- ➔ Expand to pickup two blocks and stack them

Showtime

Let’s check out all the awesome code

That's it for today

- **Shut down the JetMax arm properly**
- Wait for the screen to go blank
- Switch the arm off
- Carefully disconnect cables
- Being all parts up and hand them to the volunteers
- Remember your number, so you have the same robotic arm next time
- See you tomorrow 🎉

OAK RIDGE COMPUTER SCIENCE GIRLS

ORCS Girls

BLOUNT COUNTY PUBLIC LIBRARY

UT-BATTELLE

SMART ROBOTICS CAMP
Day 3

ORCSGirls

ORCSGirls

Review what we learned

Smart Robotics Camp

TRADITIONAL PROGRAMMING

Rules for "Is this picture a pig?"

Is it pink?
Does it have a tail?
Pointed ears?
Four legs?
And so on...

Traditional code for "Is this picture a pig?"

```
if (isPink(picture) and
hasTail(picture) and
hasPointedEars(picture)
and hasFourLegs(picture)
and …)
   print "Yes!"
else
   print "No!"
```

Smart Robotics Camp

Yes, even in Oak Ridge

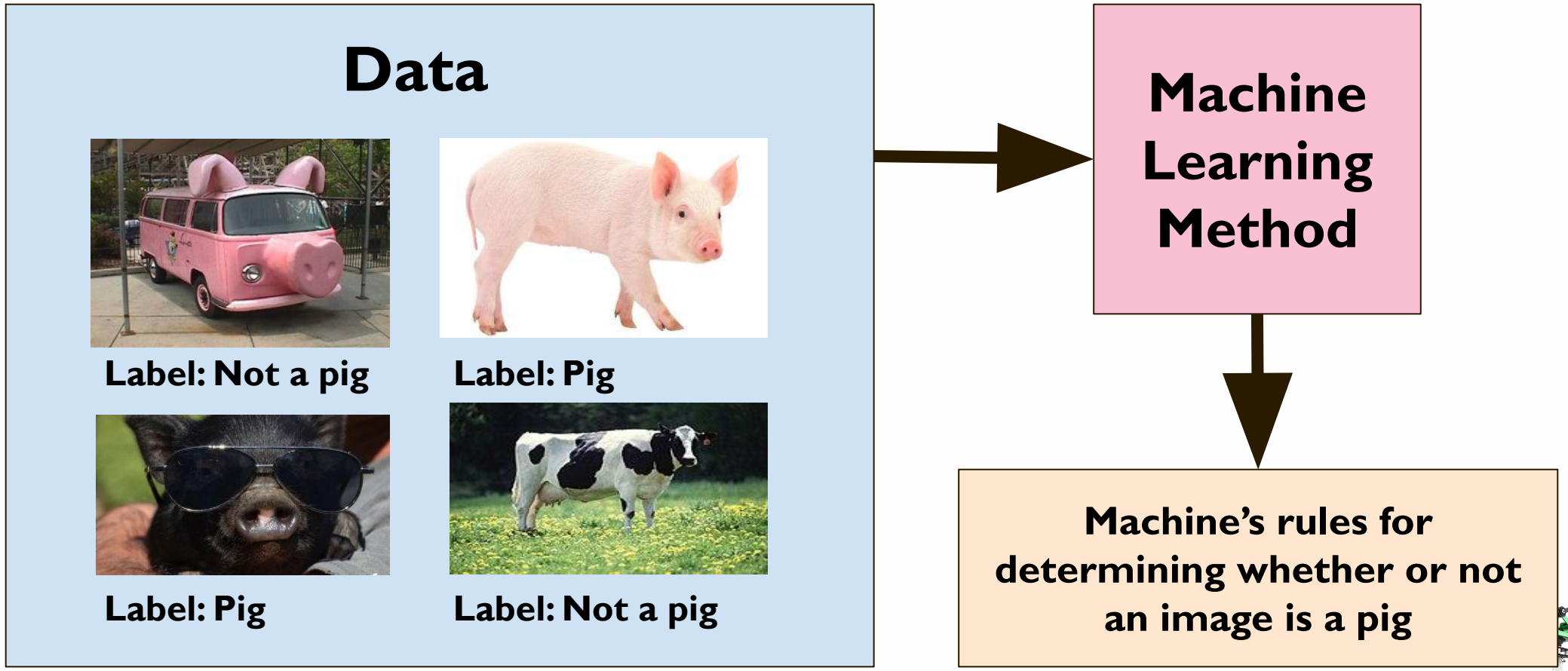
?

HOW DO YOU DECIDE WHAT THE RULES ARE?

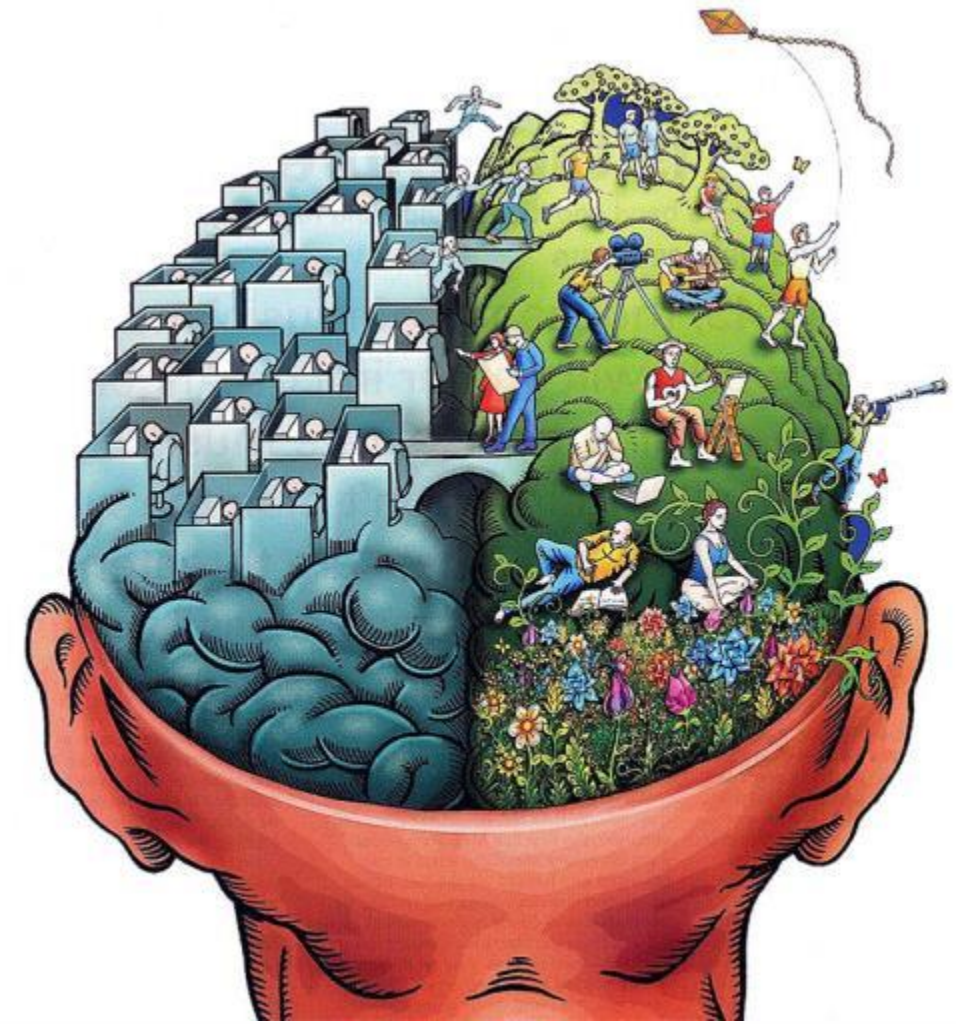
YOU DON'T!
LET THE COMPUTER DO IT WITH
MACHINE LEARNING!

MACHINE LEARNING

A machine learning method takes a bunch of data and "learns" from it!



WHAT'S THE DIFFERENCE BETWEEN MEMORIZING AND LEARNING?



Smart Robotics Camp

DID IT "LEARN" SOMETHING?

Label: Not a pig

Label: Pig

Label: Pig

Label: Not a pig

Training Data

The data we give to the machine learning method to learn from

Label: Not a pig

Label: Pig

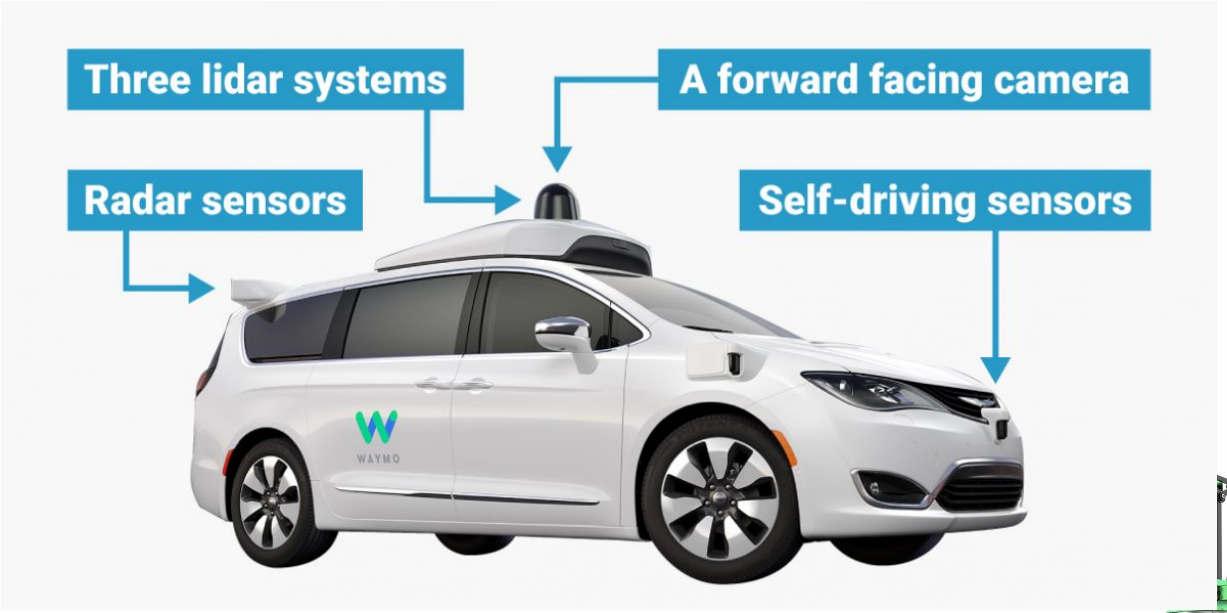
Testing Data

The data we hold out and use to check to see if the method actually learned something!

WHERE IS ML USED?

Google

amazon



Smart Robotics Camp

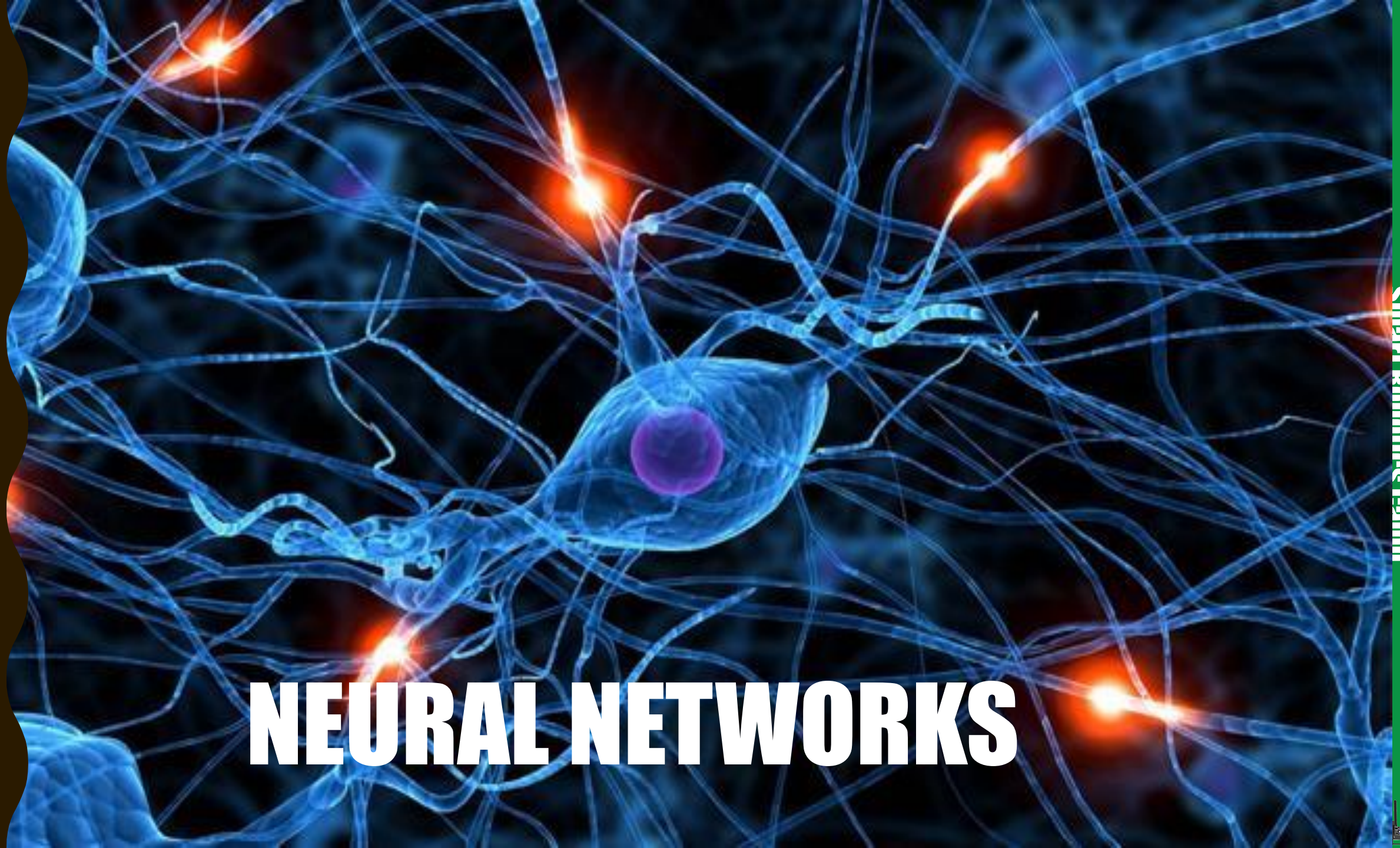
WHY DOES IT WORK NOW?

Lots of Data

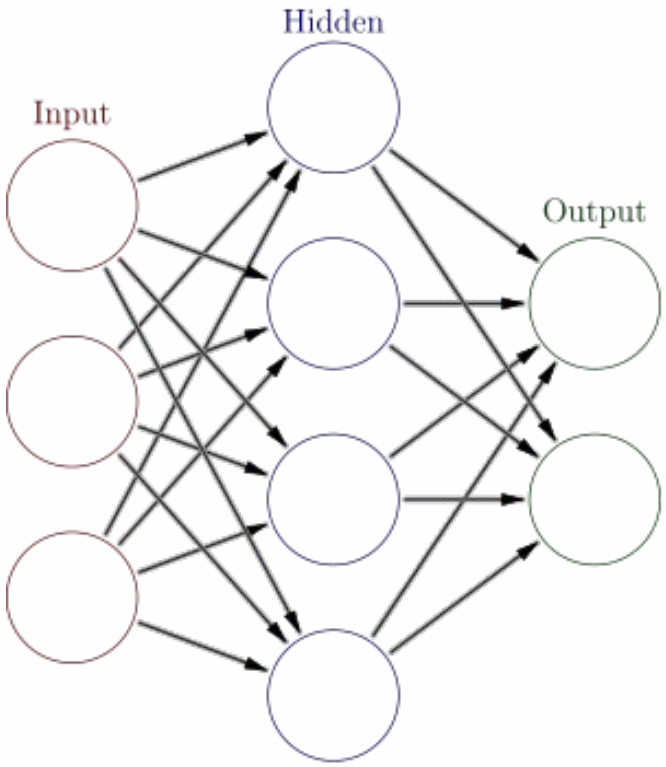
- 350 million new photos to Facebook every day
- 300 hours of video uploaded to YouTube every minute
- 500 million Tweets every day

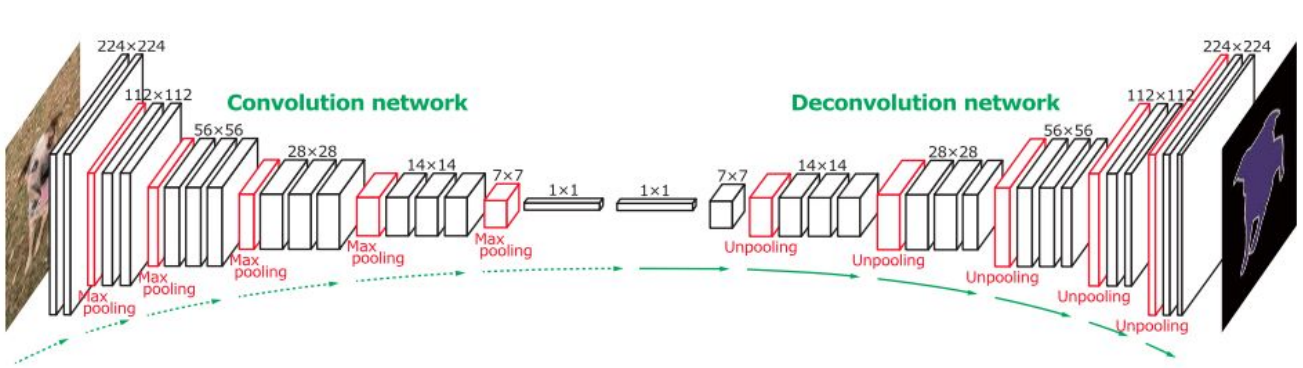
Powerful Computers

- Computers are faster and bigger than ever!


NEURAL NETWORKS

NEURAL NETWORK EXPLANATION





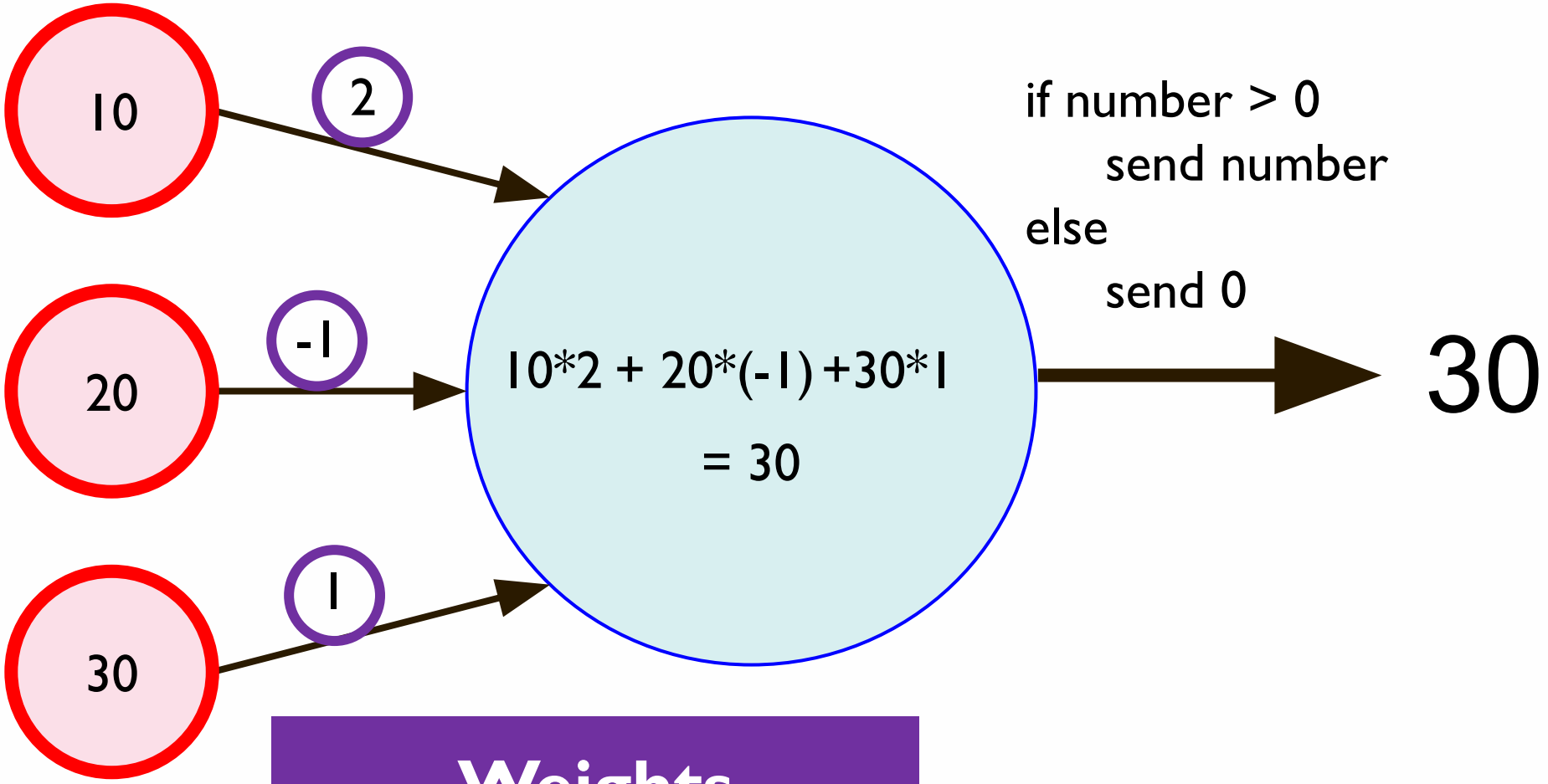
Let's use this as an example

ChatGPT-3

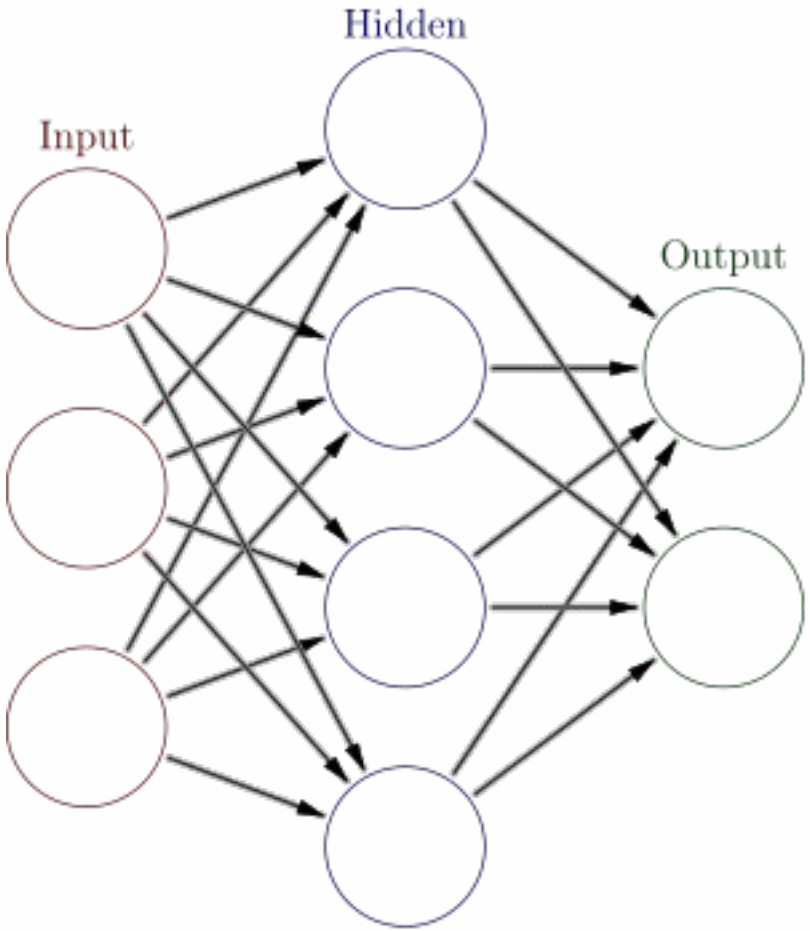
- Number of layers: 96
- Number of attention heads: 96
- Dimensions of its hidden layers: 12288
- Sequence length: 2048
- **Number of parameters: 175 Billion**

Smart Robotics Camp

NEURAL NETWORK EXPLANATION



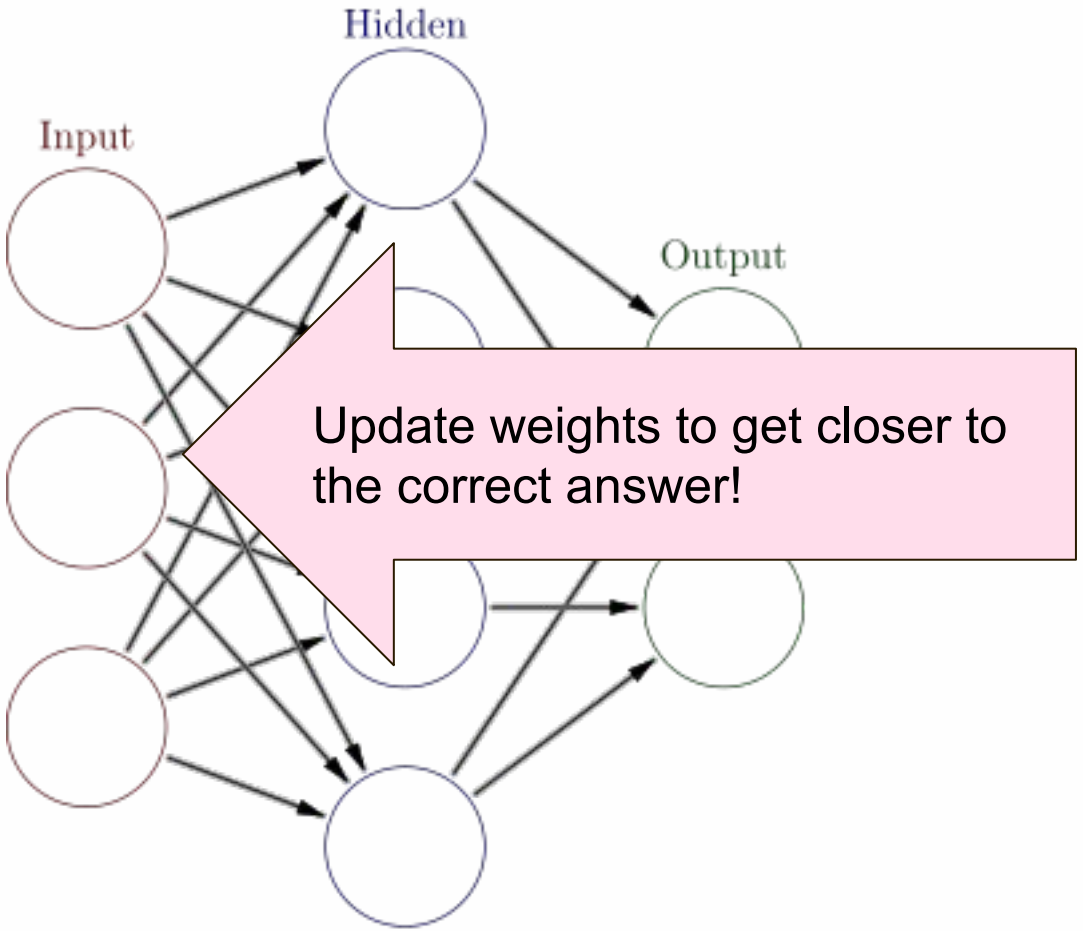
Neural Network Explanation



Guessed Answer	Correct Answer
0.3	1
0.7	0

Smart Robotics Camp

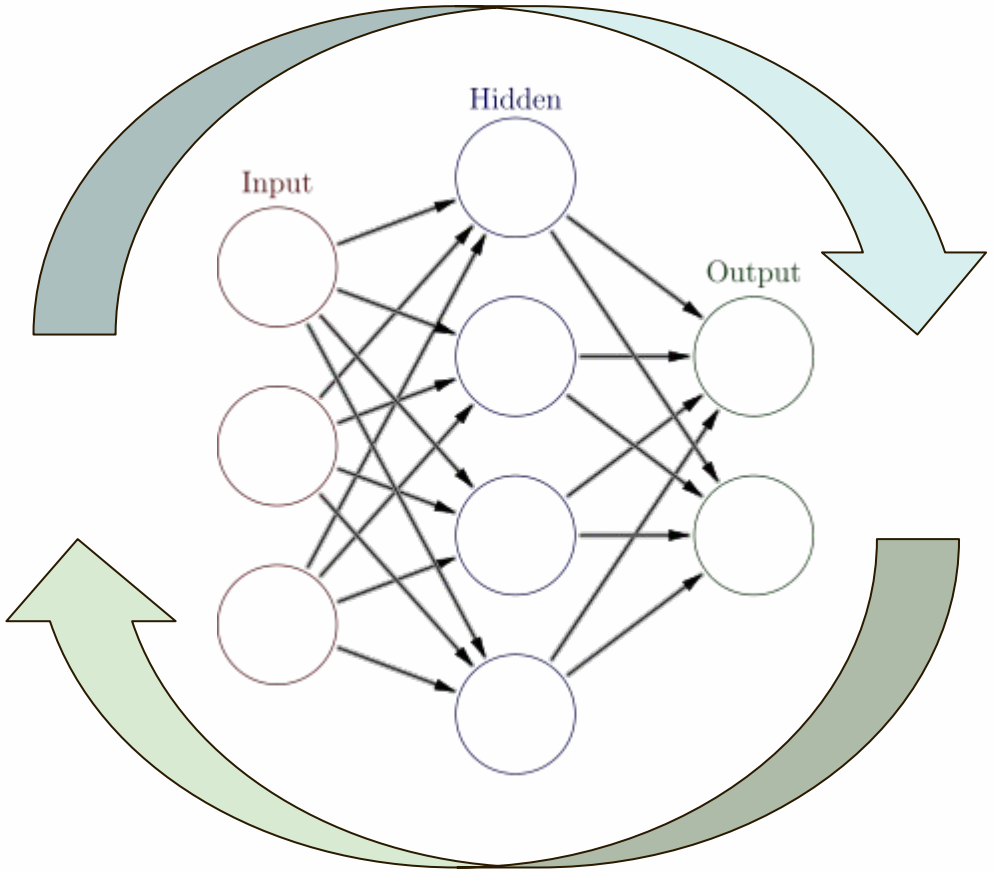
Neural Network Explanation



Guessed Answer	Correct Answer
0.3	1
0.7	0

Neural Network Explanation

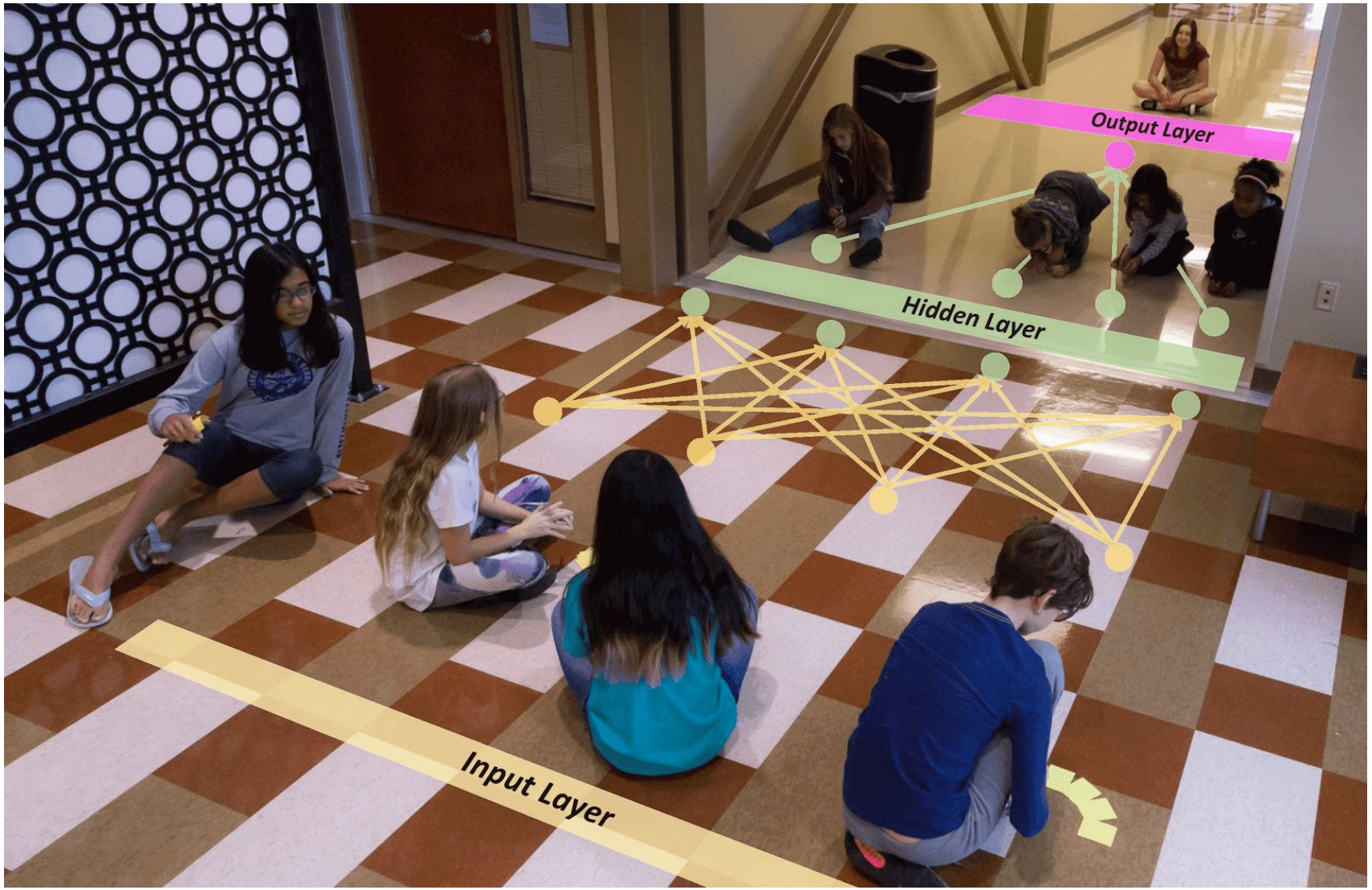
Run a data example through



Update weights

Keep repeating until your answers on the training set are good enough!

Human Neural Network Activity

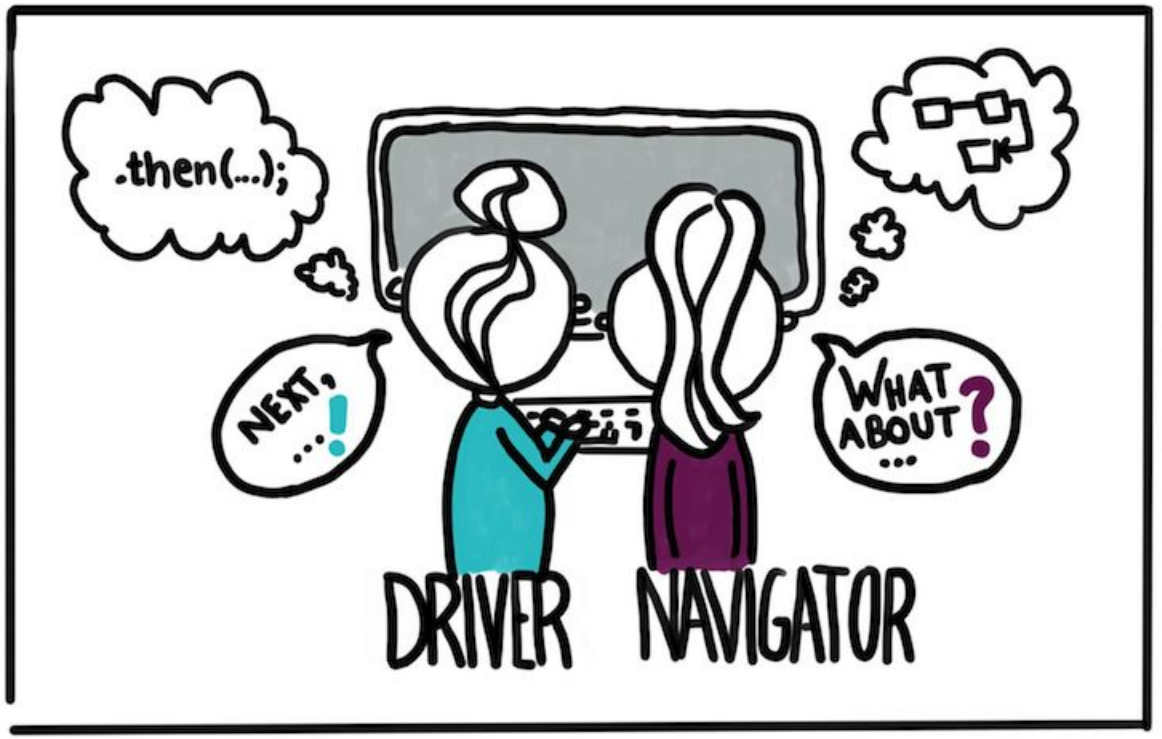


WHAT IS ALGORITHMIC BIAS?

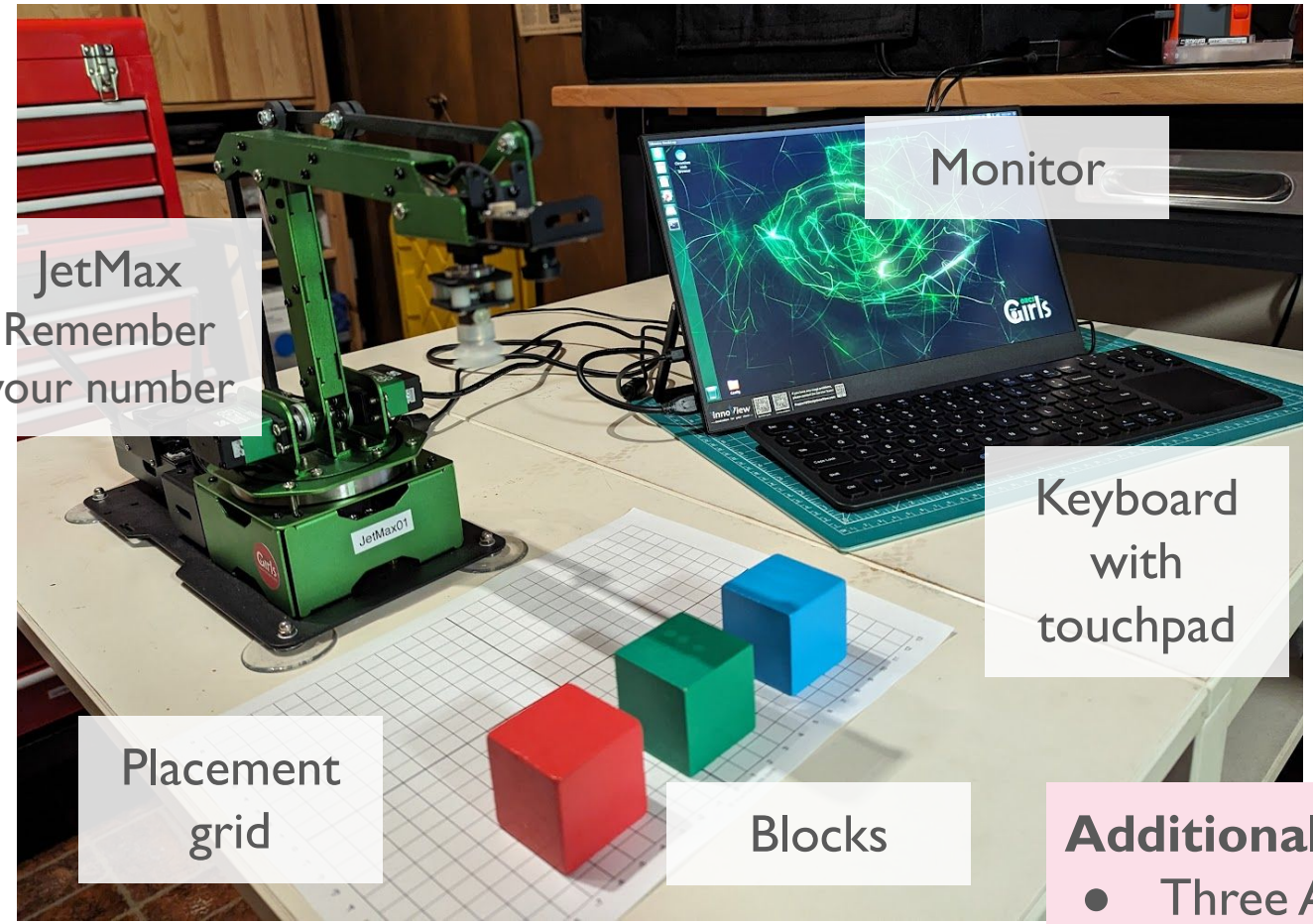
Smart Robotics Camp

Remember *Driver Navigator* rules

- Make sure everyone gets to perform every task
- Take turns in setting up
- Help each other
- Listen and communicate
- Great teamwork leads to great progress and results
- Valuable skill to learn



Setting up the JetMax



Clear your workspace

Decide who picks up what

Volunteers will hand our materials

You need

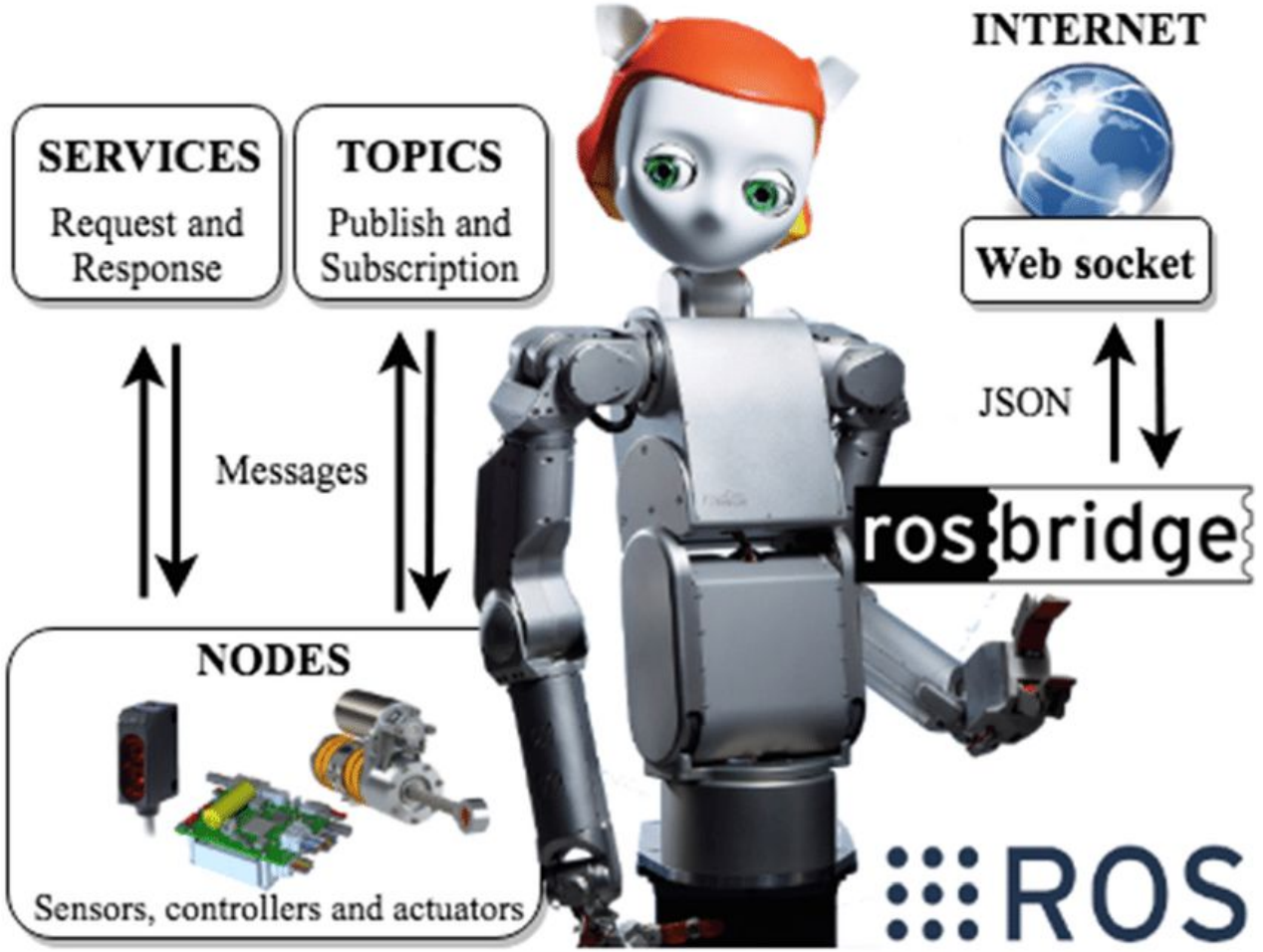
- JetMax robot arm
- **Use same robot number**
- Monitor
- Keyboard
- Placement grid
- Three color blocks
- Power supply JetMax
- Power supply monitor
- HDMI cable

Additional

- Three Apriltag blocks
- Recycling mat and cards

Smart Robotics Camp

Robot Operating System



Calibrating camera - sucker offset

Open web browser to view camera image
Open a terminal to enter commands
We will do this together

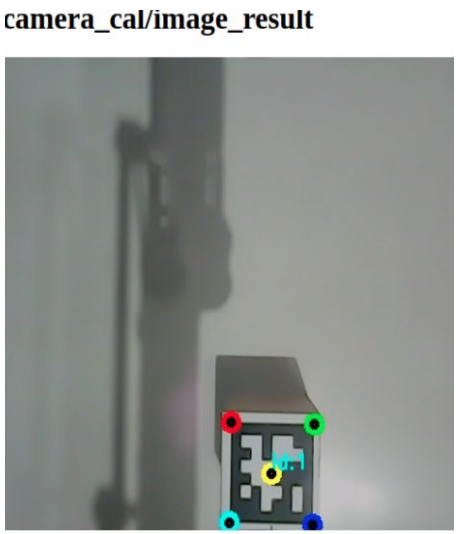
```
rosservice call /camera_cal/enter
rosservice call /camera_cal/set_running "{data: True}"
rosservice call /camera_cal/down
```

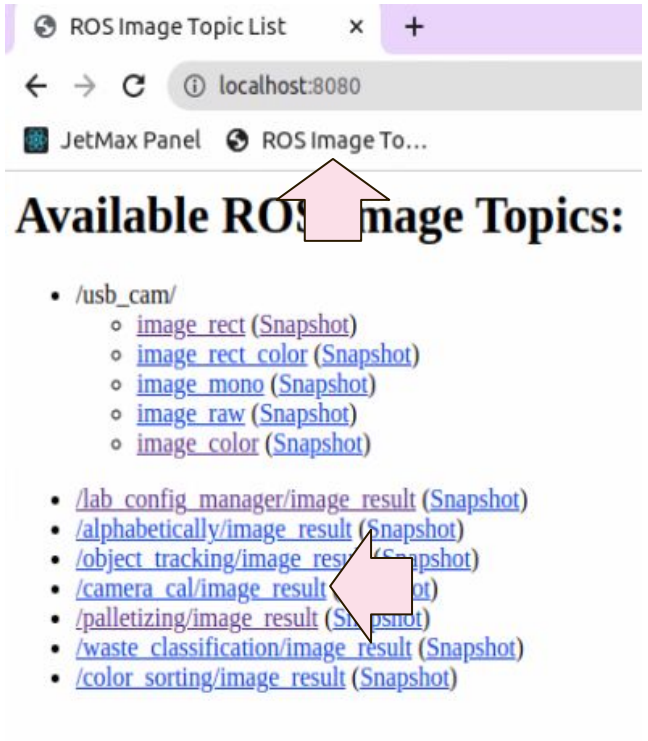
Position **Apriltag block 1** under middle of nozzle with the 1 facing the arm

```
rosservice call /camera_cal/up
```

Wait for dots to align on the video feed in the browser

```
rosservice call /camera_cal/save
rosservice call /camera_cal/exit
```

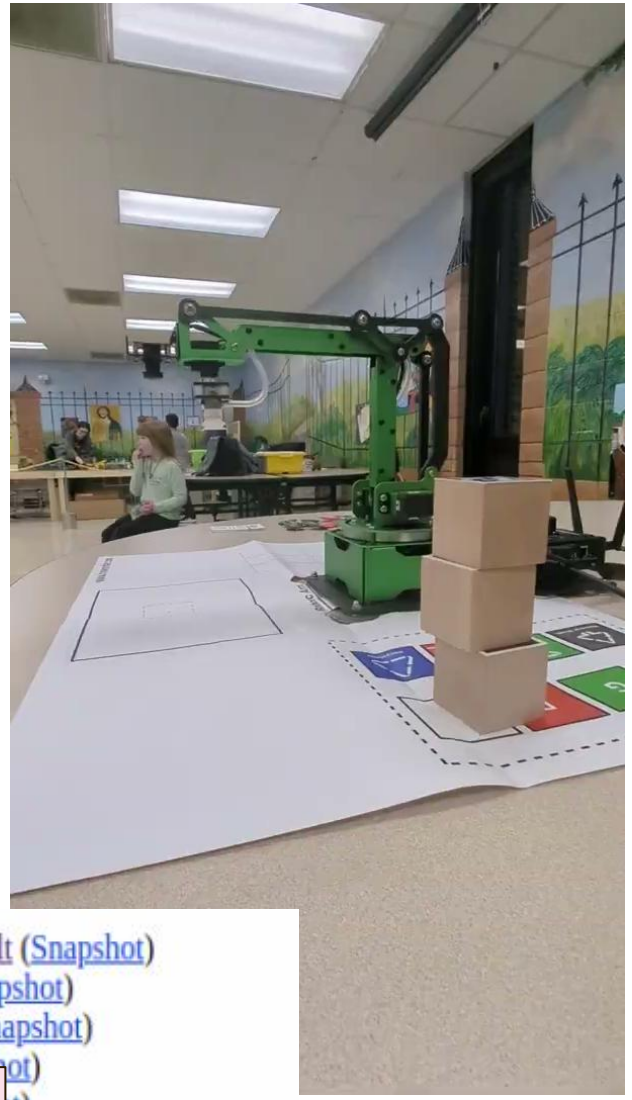




Smart Robotics Camp

Block stacking demo

Open web browser to view camera image
Open a terminal to enter commands
Make sure you have space around the arm



```
rosservice call /palletizing/enter
rosservice call /palletizing/set_running "{data: True}"
```

Place **Apriltag blocks** in front of the arm. It will pick them up and stack them. If there is more than one it will stack them in order

Once you are done, exit the demo with

```
rosservice call /palletizing/exit
```

Take turns in starting and stopping the demo and placing blocks

- /lab_config_manager/image_result (Snapshot)
- /alphabetically/image_result (Snapshot)
- /object_tracking/image_result (Snapshot)
- /camera_cal/image_result (Snapshot)
- /palletizing/image_result (Snapshot)
- /waste_classification/image_result (Snapshot)
- /color_sorting/image_result (Snapshot)

Recycling

Close your web browser to save memory
Open a terminal to enter commands

```
rosrun Ai_JetMax waste_classification.py
```

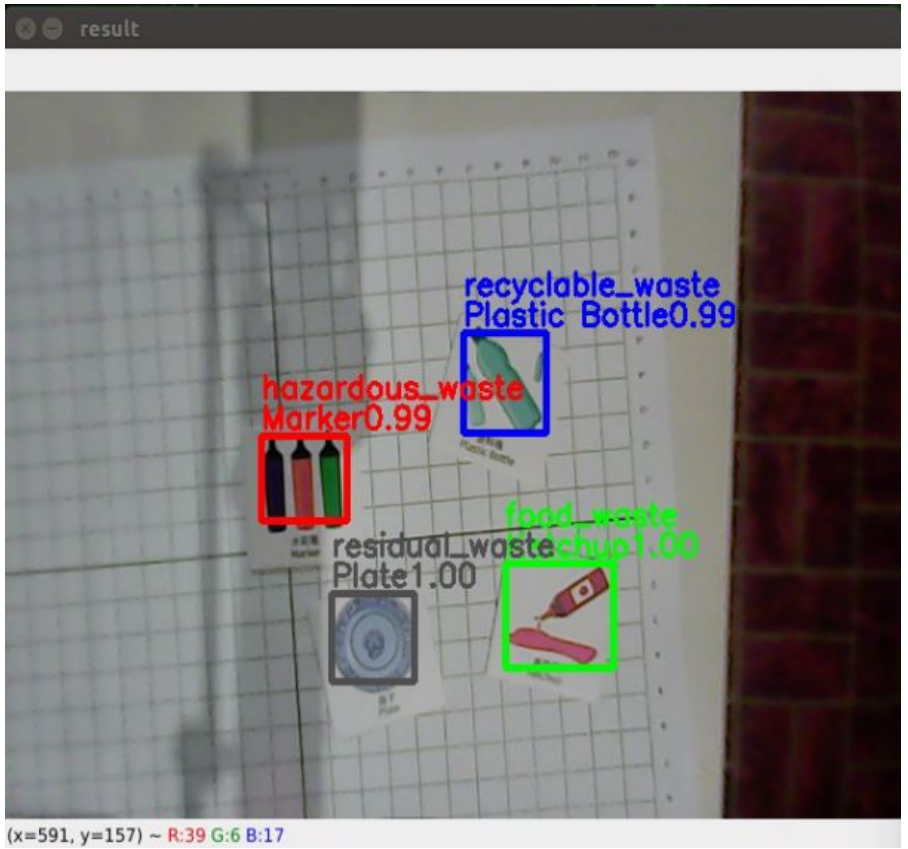
After a **long time** a window with the camera feed will appear

Place recycling cards under the arm

If will classify the waste and show confidence level

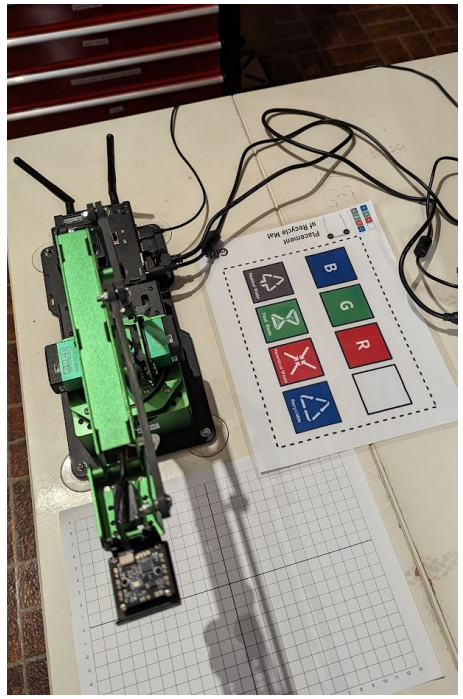
Observations
Do cards work equally well in all orientations.
Do all cards work equally well?
What if they overlap?

Exit the program by pressing Control C



Recycle sorting demo

Place the recycling mat as shown
Open web browser to view camera image
Open a terminal to enter commands



```
rosservice call /waste_classification/enter
rosservice call /waste_classification/set_running "{data: True}"
```

Place **recycling cards** in front of the arm.
It will pick them up and move the to the correct waste location

Once you are done, exit the demo with

```
rosservice call /waste_classification/exit
```

Take turns in starting and stopping the demo and placing blocks

- /lab_config_manager/image_result (Snapshot)
- /alphabetically/image_result (Snapshot)
- /object_tracking/image_result (Snapshot)
- /camera_cal/image_result (Snapshot)
- /palletizing/image_result (Snapshot)
- /waste_classification/image_result (Snapshot)
- /color_sorting/image_result (Snapshot)

Smart Robotics Camp

That's it for today

- **Shut down the JetMax arm properly**
- Wait for the screen to go blank
- Switch the arm off
- Carefully disconnect cables
- Being all parts up and hand them to the volunteers
- Remember your number, so you have the same robotic arm next time
- See you tomorrow 🎉

OAK RIDGE COMPUTER SCIENCE GIRLS

ORCS Girls

BLOUNT COUNTY PUBLIC LIBRARY

UT-BATTELLE

SMART ROBOTICS CAMP
Day 4

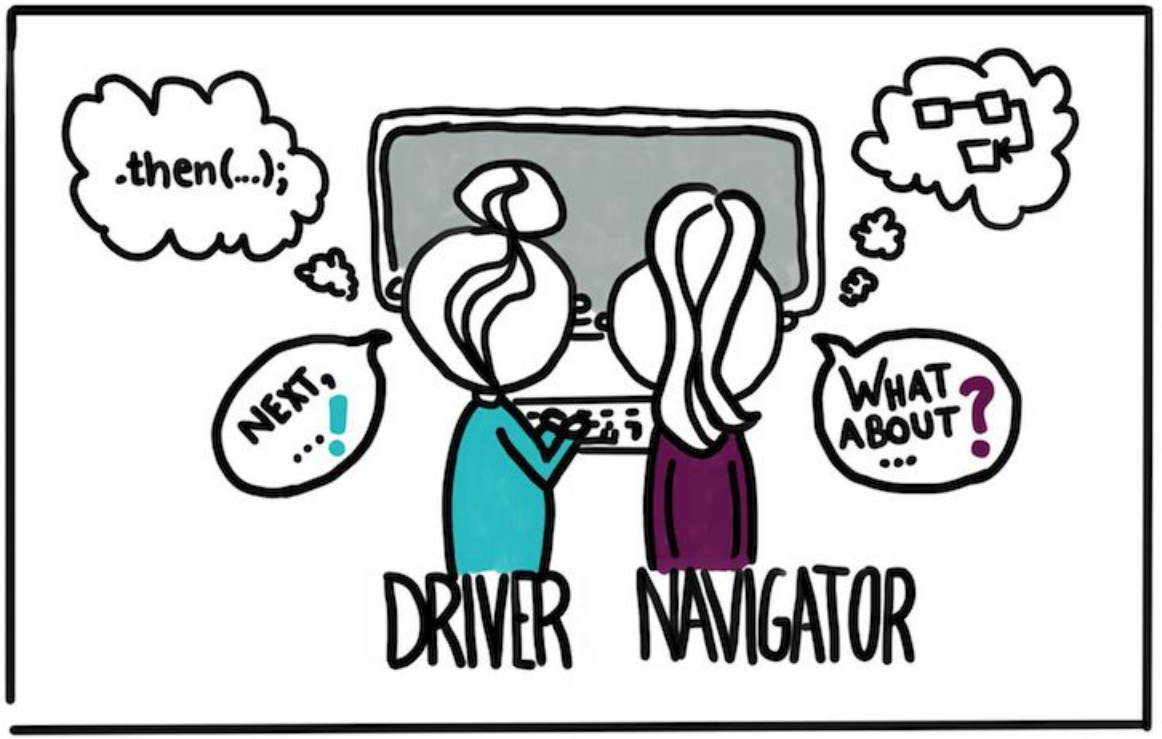
ORCSGirls

ORCSGirls

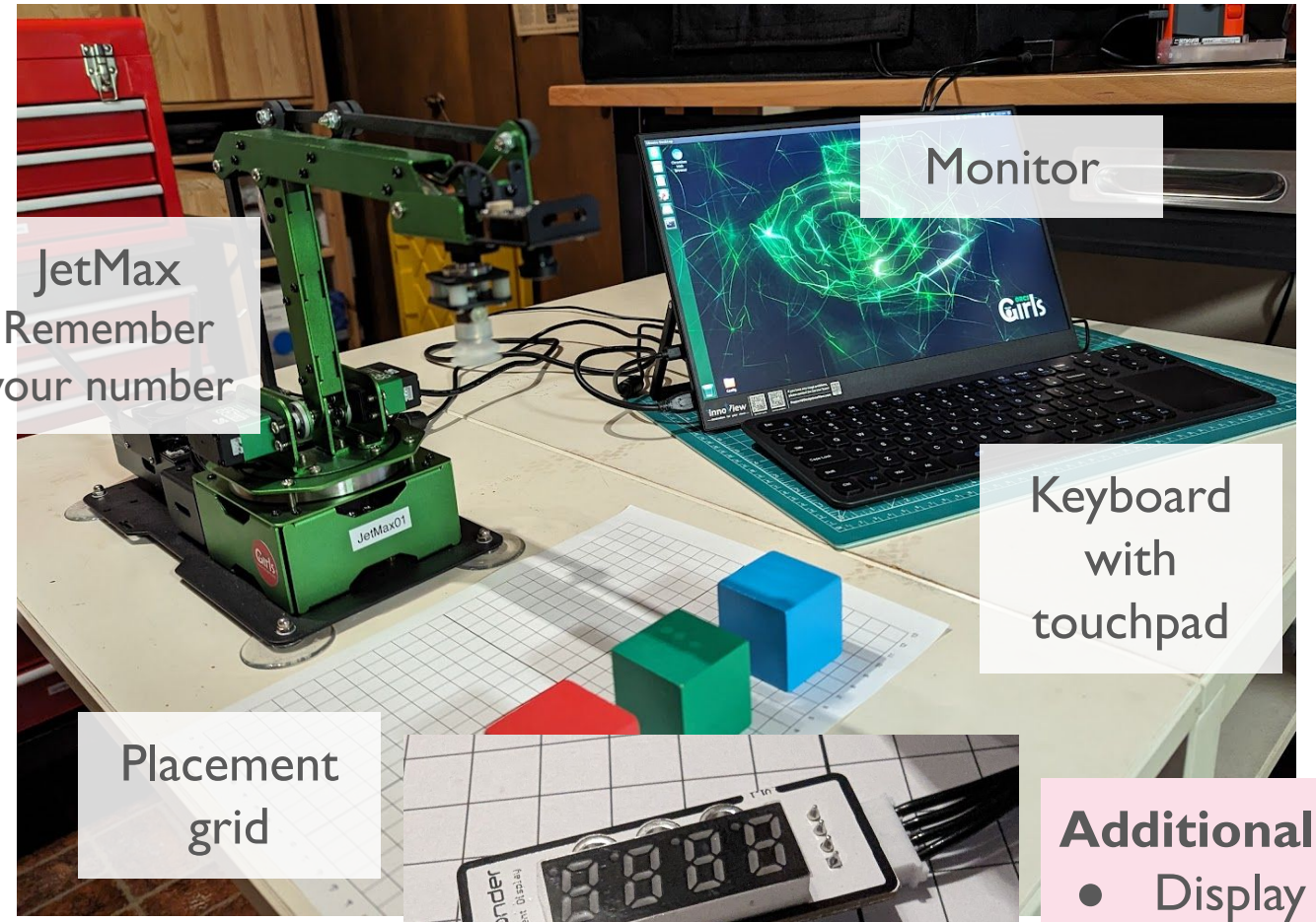
Review what we learned

Remember *Driver Navigator* rules

- Make sure everyone gets to perform every task
- Take turns in setting up
- Help each other
- Listen and communicate
- Great teamwork leads to great progress and results
- Valuable skill to learn



Setting up the JetMax



Clear your workspace

Decide who picks up what

Volunteers will hand our materials

You need

- JetMax robot arm
- **Use same robot number**
- Monitor
- Keyboard
- Placement grid
- Three color blocks
- Power supply JetMax
- Power supply monitor
- HDMI cable

Additional

- Display module and cable
- Connect **before** starting the arm

Smart Robotics Camp

Connecting the display

- Connect the display to the JetMax as shown
- Connectors will only go in in one direction - be gentle
- After the display is connected start up the JetMax

Controlling the display in Python

```
import hiwonder
import time

mtx = hiwonder.TM1640(4, 9)

mtx.brightness(4)

mtx.tube_display_int(10)

mtx.tube_display_float(3.14)

mtx.refresh()
```

Refresh is needed to actually update the display

Display Commands

- Open a terminal
- Go to the Python folder
- Run the **tube.py** program
- Edit the program and modify the code - e.g. count higher, slower

- **Can make a countdown showing 0.1 second steps?**

Showtime

Let’s checkout all the cool displays 🔢

Color cube demos

Open a terminal

Go to the color folder **`cd Python/color`**

Run the demos with

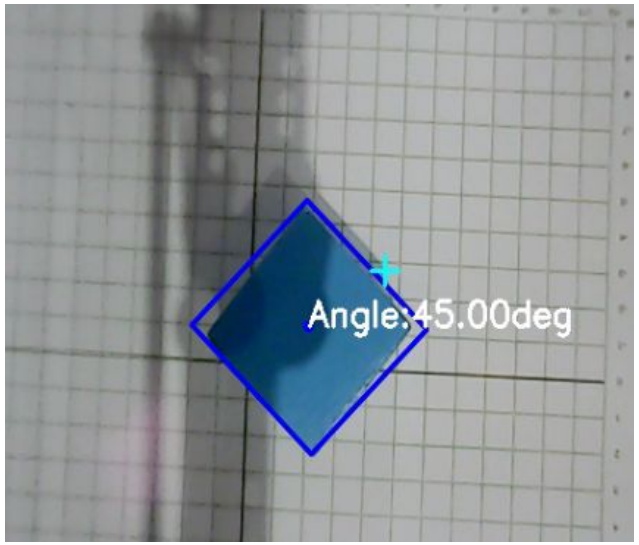
`python3 color_angle.py`

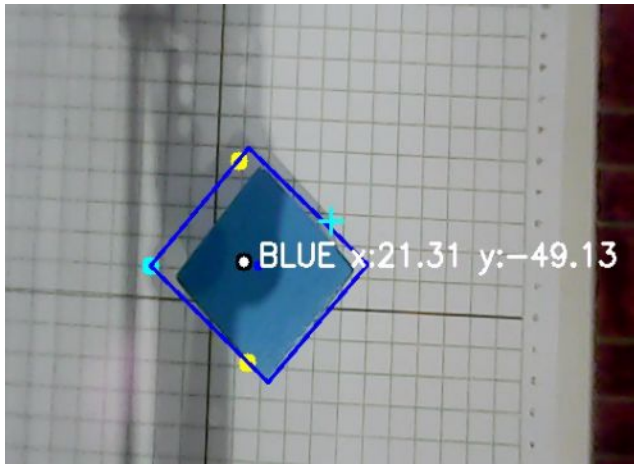
or

`python3 color_position.py`

Move or rotate the block and observe the numbers

To exit either code, press **Control C**





```python
def image_proc(img):
    img_h, img_w = img.shape[:2]
    frame_gb = cv2.GaussianBlur(np.copy(img), (5, 5), 5)
    frame_lab = cv2.cvtColor(frame_gb, cv2.COLOR_RGB2LAB)  # Convert rgb to lab
    blocks = []
    for color_name, color in state.target_colors.items():  # Loop through all selected colors
        frame_mask = cv2.inRange(frame_lab, tuple(color['min']), tuple(color['max']))
        eroded = cv2.erode(frame_mask, cv2.getStructuringElement(cv2.MORPH_RECT, (3, 3)))
        dilated = cv2.dilate(eroded, cv2.getStructuringElement(cv2.MORPH_RECT, (3, 3)))
        contours = cv2.findContours(dilated, cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_NONE)[-2]
        contour_area = map(lambda c: (c, math.fabs(cv2.contourArea(c))), contours)
        contour_area = list(filter(lambda c: c[1] > 1000, contour_area))  # Eliminate contours that are too small

        if len(contour_area) > 0:
            for contour, area in contour_area:  # Loop through all the contours found
                rect = cv2.minAreaRect(contour)
                center_x, center_y = rect[0]
                box = cv2.boxPoints(rect)  # The four vertices of the minimum-area-rectangle
                box_list = box.tolist()
                box = np.int0(box)

                cv2.drawContours(img, [box], -1, hiwonder.COLORS[color_name.upper()], 2)
                cv2.circle(img, (int(center_x), int(center_y)), 1, hiwonder.COLORS[color_name.upper()], 5)
                angle = rect[2]
                angle = angle - 90 if angle > 45 else angle
                s = "Angle:{:0.2f}deg".format(angle)
                cv2.putText(img, s, (int(center_x), int(center_y)), cv2.FONT_HERSHEY_SIMPLEX, 0.8, (255, 255, 255))
                print(s)

    cv2.line(img, (int(img_w / 2 - 10), int(img_h / 2)), (int(img_w / 2 + 10), int(img_h / 2)), (0, 255, 255), 2)
    cv2.line(img, (int(img_w / 2), int(img_h / 2 - 10)), (int(img_w / 2), int(img_h / 2 + 10)), (0, 255, 255), 2)

    return img

def image_callback(ros_image):
    image = np.ndarray(shape=(ros_image.height, ros_image.width, 3), dtype=np.uint8, buffer=ros_image.data)
    frame_result = np.copy(image)
    frame_result = image_proc(frame_result)
    image_bgr = cv2.cvtColor(frame_result, cv2.COLOR_RGB2BGR)
    cv2.imshow("result", image_bgr)
    cv2.waitKey(1)


if __name__ == '__main__':
    rospy.init_node(ROS_NODE_NAME, log_level=rospy.DEBUG)
    state = State()
    state.load_camera_params()
    if state.camera_params is None:
        rospy.logerr("Can not load camera params")
        sys.exit(-1)
    jetmax = hiwonder.JetMax()
    jetmax.go_home()
    rospy.sleep(1)
    state.target_colors = rospy.get_param('/lab_config_manager/color_range_list', {})
    del[state.target_colors['white']]
    del[state.target_colors['black']]
    del[state.target_colors['ball']]
    del[state.target_colors['tennis']]
    image_sub = rospy.Subscriber('/usb_cam/image_rect_color', Image, image_callback)
    try:
        rospy.spin()
    except KeyboardInterrupt:
        sys.exit(0)
```

Main program starts here

Color Angle Code 😮

The code subscribes to the camera feed - we can check it on the webpage.

When there is a new frame the **image_callback** routine is called

image_callback gets the data and calls **image_proc** to determine the block angle and annotate the image

Where is the angle printed on the screen - shout out the line number

Smart Robotics Camp

Display the block angle

Somewhere in the main program we need to add these lines to create the display object **mtx**

```
mtx = hiwonder.TM1640(4, 9)
mtx.brightness(4)
```

In **image_proc** where the print(s) statement is, we add these lines to display the angle

```
mtx.tube_display_float(angle)
mtx.refresh()
```

Let's go

- Open a terminal
- Go to the Python/color folder
- Make a copy of color_angle.py

```
cp color_angle.py my_color_angle.py
```

- Edit the new file and to add the code to display the angle on the display

Showtime

For the last time

- **Shut down the JetMax arm properly**
- Wait for the screen to go blank
- Switch the arm off
- Carefully disconnect cables
- Being all parts up and hand them to the volunteers

Group photo and graduation

Want more?

Sign up for classes
www.orcsgirls.org/classes

	Virtual Classes
Jun 29	Computer Simulations for Science ⭐⭐
Jul 27	Introduction to JavaScript ⭐⭐
Aug 10	Did someone say Cookies? ⭐⭐

More in person and virtual classes coming soon ..

Smart Robotics Camp

ORCSGirls Junior

Sign up for classes
www.orcsgirls.org/junior

Jul 22-25

Scratch Camp⭐
Virtual

ORCSGirls Junior
Director
Katie Bates

Open to girls in
grades 2-4.

Smart Robotics Camp

Smart Robotics Camp

Big thank you to our volunteers, partners and sponsors.

UT-BATTELLE

BLOUNT COUNTY PUBLIC LIBRARY

SOUTHLAND BOOKS & CAFE

ORCS Girls

www.orcsgirls.org

contact@orcsgirls.org

ORCSGirls

ORCSGirls